artificial intelligence, data mining
sigma-if neural network, decision space selection,
non-destructive neural network pruning

Maciej HUK*

THE SIGMA-IF NEURAL NETWORK AS A METHOD OF DYNAMIC SELECTION OF DECISION SUBSPACES FOR MEDICAL REASONING SYSTEMS

To-date research in the area of applied medical artificial intelligence systems suggests that it is necessary to focus further on the characteristic requirements of this research field. One of those requirements is related to the need for effective analysis of multidimensional heterogeneous data sets, which poses particular difficulties when considering AI-suggested solutions. Recent works point to the possibility of extending the activation function of a perception to the time domain, thus significantly enhancing the capabilities of neural networks. This change results in the ability to dynamically tune the size of the decision space under consideration, which stems from continuous adaptation of the interneuron connection architecture to the data being classified. Such adaptation reflects the importance of individual decision attributes for the patterns being classified, as defined by the Sigma-if network during its training phase. These characteristics enable effective employment of such networks in solving classification problems, which emerge in medical sciences. The described approach is also a novel, interesting area of neural network research. This article discusses selected aspects of construction as well as training of Sigma-if networks, based on a sample problem of classifying Arabic numeral images.

1. INTRODUCTION

The development of medical advisory systems has always been linked to broadening the means of patient data acquisition, as well as creating new ways of processing and utilizing such data. The rapid abandonment of formal methods in favour of heuristic data analysis enabled developers to widen the scope of data subject to processing and to narrow the gap between artificial diagnostic systems and medical practitioners [19, 32, 33]. The recent years in particular have been a period of unprecedented advancements in data acquisition and processing techniques; yet the ever greater demands placed on medical advisory systems necessitate inventing new methods of data processing, aimed specifically at medical uses.

The specificity of medical data results from numerous factors, including, but not limited to, completeness, consistency and accuracy criteria. Of importance are also geographical and cultural aspects; however the greatest problem facing developers of medical information systems is the considerable heterogeneity of medical data, and the

* Department of Computer Science, Wroclaw University of Technology, Maciej.Huk@*pwr.wroc.pl*.

associated difficulties in determining the relevancy of individual information classes - or even individual records. [7]

Considering the heterogeneity of medical data (such as X-ray and EMR images, ECG and EEG readouts, information on age, weight and size of the patient as well as on his/her family and environment), it is often difficult to judge which factors play a key role in diagnosing specific diseases and conditions. Most existing approaches to automatic reasoning using artificially-limited search spaces often lead to classification errors resulting from the lack of completeness in patient data files. This drawback calls into question the application of computerized systems for automatic diagnostics and treatment. Alternative solutions, which involve presenting the system with a full dossier of patient data, frequently introduce high levels of unwanted noise and redundant information [15].

Similar difficulties emerge even when processing minute fragments of patient data. A good example here is medical image processing. Such images often contain a large amount of noise interspersed with valuable information. Automated image analysis is typically linked to the application of neural networks, albeit with no clear set of reasoning guidelines or attributes on which such analysis relies [3, 32]. The conclusions presented below suggest, however, that it is in fact possible to modify typical neural networks, trained with the use of back propagation algorithms, in a way which would allow us to assess the importance of individual attributes affecting classification. This enables subsequent verification of conclusions derived by the system without the need to resort to additional tools for extracting knowledge from neural networks [10, 16].

2. THE SIGMA-IF NEURAL NETWORK

The basic constituent of a traditional neural network is the perceptron, which processes incoming signals from interneuronal connections by means of two functions: the activation function *A* and the output function *F*. The former determines the activation level of the neuron, while the latter formulates an answer which is then communicated to other areas of the network. The importance of the output function as well as of weights attached to individual connections has been thoroughly analyzed in numerous publications [1, 4, 8, 10, 11, 25]. It is worth noting, however, that the activation function is almost always assumed to be a linear combination of input values, which is not supported by theory. Analyzing the behaviour of neurons with various activation functions can therefore lead to new processing structures, with novel, useful characteristics [2, 6, 16, 22, 23].

The Sigma-if neuron (fulfilling the above criteria) has its dendrites divided into K classes, which implies associating a special parameter θ with each network connection, describing its class participation. All these parameters together form the class vector Θ. The number of different classes of input connections K is set by network learning method and is limited to a selected value M.

This construct can be clearly interpreted from a biological standpoint: in the case of real neurons, individual dendrites differ in length, which means that data transport is neither instantaneous nor correlated within any particular connection. This phenomenon forms one of the characteristics which allow a real neural network to associate incoming signals with particular connections and processing areas. Hence, the classes of input connections of an artificial neuron acquire a representation of length, or – more intuitively – the transfer time for signals through interneuronal connections. In light of this fact, *Θ* will hereafter be called the delay vector, and its individual values will be treated as delays introduced by particular connections.

We can thus alter the characteristics of determining the activation of a neuron – from instantaneous to time-driven. Just as in the case of a classical perceptron, this process entails the accumulation of input signals, multiplied by individual components of the weight vector *w*. However, in our case, the accumulation consists of distinct stages. The activation of the neuron, *net*, is increased in a stepwise fashion by sums of incoming signals (in the order determined by their respective delays). This process continues until the *net* exceeds the neuron's activation threshold *net** - subsequently, the output value is determined and all late signals (ones which still haven't arrived) are disregarded. If the activation threshold is not achieved even after all input signals are integrated, the output remains zero.

We can easily surmise that class k of dendrites contributes a value of *Δk* to the activation of the neuron, where:

$$\Delta k = \sum_{i=1}^{M} w_i \, x_i \delta(k, \theta_i) \tag{1}$$

and δ(k,θ) is Kronecker's delta:

$$\delta(k,\theta) = \begin{cases} 0 : k \neq \theta \\ 1 : k = \theta \end{cases}. \tag{2}$$

Given *Δk*, the behaviour of the neuron can be described by deriving a recursive expression of the following form:

$$net(k) = \Delta k \cdot H(net * - net(k-1)) + net(k-1) \tag{3}$$

where:

$$net(k) = 0 \;\; for \; k \not\in \{1..K\}, \tag{4}$$

*net** is a constant and *H* is Heaviside's function. The activation of a processing unit of type Sigma-if (conditional sum) is then expressed as:

$$A = \begin{cases} net(1) : net(1) \geq net * \\ 0 \qquad : net(1) < net * \end{cases} \tag{5}$$

while its output value y is equal to:

$$y = F(A(w,\ x))\,. \qquad (6)$$

The described mechanism, given the right distribution of delays and weights, allows the discretization of the data space into a number of input parameters, dynamically adjusted to the type of patterns emerging at the input of the network. Individual neurons may attempt to undertake a decision for a set input value K times, initially dissecting the data space by hypersurfaces with a small number of dimensions, then – if the need arises – increasing their complexity. The grouping of dendrites into classes, suitable for a particular task, allows us to eliminate parameters which are either insignificant or detrimental to the final result. Decreasing the number of dimensions of the search space makes it easier for the network to classify data while preserving the original number of constitutent neurons (i.e. the hypersurfaces dissecting the data space). In a special case all dendrites belong to the same delay class, the Sigma-if network is equivalent to a perceptron network.

Another possible interpretation relates the described technique to dynamic neural network pruning [16]. The approach is, however, a highly specific method of minimizing network size, in that no connections or neurons are actually removed from the network [compare: 23, 28]. Given the right distribution of delays, connections are used only when they prove necessary for the system to reach a decision. In spite of the synchronous mode of operation, the network is actually asynchronous in nature. At the same time it becomes far easier to train and operate.

What is even more interesting; the activation threshold of the output function does not impose any limits on network operation and does not introduce artificial conditions involving individual input parameters. This is most likely due to the fact that regardless of the input values, for a given distribution of delays the BP algorithm can select suitable weights to ensure optimal network operation [21, 24].

3. DELAY VECTOR SELECTION

All the above mentioned characteristics of the Sigma-if network are strictly dependent on proper selection of values for individual elements of the delay vector, adjusted to the problem being considered by the network. Hence, a special variant of the back propagation algorithm has been developed, allowing for the selection of individual delay times during the training phase. This variant is analogous to the simulated annealing algorithm with two guiding parameters, where the particles are represented by interneuronal connections. Energy is equivalent to the neural network error function (its gradient governs the annealing process), while the guiding parameter set is composed of connection weights and introduced delays [16].

The BP algorithm only adjusts the weights of those dendrites (active connections), which, in the preceding pass of the training process, contributed to the activation of their respective neurons *and* compounded the overall network error. To some degree, this procedure can counteract the destruction of weights when the network is presented with different classes of training patterns than the current one. Given the right distribution of delays, each class can be recognized using different input parameters. In order to come up with the proper delays, it is necessary to adequately modify the active connections during each pass of the BP algorithm:

- *increasing* and *decreasing* them for active connections which contribute (within their respective classes) to neuronal activation to the *smallest* and *greatest* degree respectively, when the overall network error decreases.
- *decreasing* them for inactive connections in the current pass of the BP algorithm, when the overall network error decreases too slowly or increases.

In this way, the training algorithm attempts to establish – for each neuron – the proper number of classes and their associations with particular connections. This procedure is beneficial, because one of the reasons why BP algorithms may prove unable to minimize error is the lack of access to some input parameter (when the connection delay is too great). In such a situation, we should extend the class of active connections with selected inactive connections, in search for the missing element necessary for further minimization of network error. On the other hand, if the minimization progresses well, we can try to eliminate from the decision process those connections which do not significantly affect neuronal activation (by moving them to less important classes) as well as whole classes, by moving highly active dendrites to classes with more significance.

4. EXPERIMENTS CONDUCTED

The operation and properties of the described approach will be presented on the basis of comparing its results with those of a classical neural network for a sample categorization of arabic numeral images. The sample networks constituted of 49 entries and 10 binary output nodes. Each entry corresponded to one pixel of the analyzed image, while the output nodes were associated with subsequent numbers (0 to 9). There was one hidden layer, fully connected to both the input and output layers. The classical network was trained using the back propagation algorithm with no momentum. As a baseline, changes of delays in the Sigma-if network occurred between the hidden layer and the input layer only. During training, sample numeral images from Fig.1 were used.

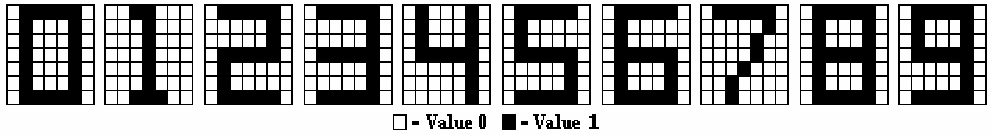

Fig. 1. Sample training data for neural networks.

The training times t, expressed in the number of epochs, show that the Sigma-if network takes more than 2 times longer to train (2.2 times to be exact) than its classical counterpart [16]. This results from the correlation between the pace of changes in network error *E* and the average neuronal activation. For the Sigma-if network, the decreased number *d** of binary connections participating in training results in neuronal activations being proportionally smaller. In order to enable the reception of similar output signals in such circumstances, it is necessary to increase the weights of active connections. Considering the constant nature of each training step, this must be "paid for" by additional training epochs, as expressed by the following empirical dependencies (*n* – neuron number; *M* – number of its inputs):

$$\frac{\partial E_n}{\partial t} \sim d_n \tag{7}$$

and:

$$\frac{\partial E^*}{\partial E} \sim \frac{\sum_n d_n^*}{\sum_n M} \tag{8}$$

The observed growth in time *t** of Sigma-if network training as compared with time *t* for its classical counterpart matches the calculated average number of active connections per hidden layer neuron (17.4), which, given 100 connections in the input layer and no more than 490 connections in the hidden layer, results in a decrease of the total number of active dendrites compared with a full network of connections, by a factor of 2.15

$$t \sim \frac{\Delta E}{\frac{\partial E}{\partial t}} \wedge \Delta E = \Delta E^* \Rightarrow \frac{t^*}{t} \sim \frac{\partial E}{\partial E^*}. \tag{9}$$

The extended training time is, however, offset by improving other characteristics of the network. If we analyze the system's capability for generalization, i.e. the ratio of correct answers for noisy data measured against the amount of noise, we find that the Sigma-if network performs better by 7.6% (on average).

Increased resistance to noise is a simple consequence of decreasing the number of input parameters considered by the proposed network during the decision process, as well as of uniform distribution of errors throughout the full spectrum of attributes. By limiting input perception to selected parameters, the network "overlooks" noise in all other attributes, thus achieving better focus on signal data.

Decreasing the number of decision attributes, however, introduces drawbacks as well. If one or more of the selected attributes is noised, the Sigma-if network proves more error-prone than its classical counterpart. This translates into an increased amount of incorrect answers (by 13% on average). We should note the near-elimination of cases in which the network makes no decision at all. Its behaviour is far more categorical than that of the classical network, due to the simplified decision space.

Aside of the possibility of utilizing the described conditional activation function for nondestructive neural network minimization, it has one more interesting area of use. Thanks to the ability to interpret classes of dendrites as determinants of their usefulness in the decision process, it is possible to incorporate the Sigma-if network in the process of deriving knowledge from data. By treating the contents of the previously presented set of numeral images as training data, we can also determine the relevancy of individual attributes and select a subset which is sufficient for classification. This reduction in the search space area can simplify subsequent phases of KDD.

5. SUMMARY

Wrapping up the presentation of conditional activation and the Sigma-if network which bases on it, we should once more recall its most important characteristics. Compared to classical neural networks, the Sigma-if approach is characterised by greater generalisation capabilities, albeit it also commits more errors. Owing to the lesser number of utilized interneuronal connections, the training time increases in inverse proportionality to this decrease. This, however, has a positive effect in that the resultant system operates some 10% faster (the actual gain is somewhat greater, but it is partially offset by the overhead introduced by additional processing structures).

Even more importantly, the presented modification of standard neural networks points to an interesting new area of research on methods of extracting knowledge from databases [20,26,27,31]. It attempts to unify the versatility and flexibility of neuronal classification with the clarity and transparency of rule-based reasoning systems [2, 9, 13, 30]. Combined with the possibility of selecting classification-relevant elements of heterogeneous data, these properties may constitute the first step towards creating methods of data analysis uniquely suited to medical uses (the image processing methods presented in the article may well be extended to other types of data) [16,18,27].

BIBLIOGRAPHY

[1] DUCH W., JANKOWSKI N., Survey of neural transfer functions, Neural Computing Surveys, Vol. 2, pp. 163–212, Santa Cruz, 1999.

[2] DUCH W., JANKOWSKI N., Transfer functions: hidden possibilities for better neural networks, Proc. 9th European Symposium on Artificial Neural Networks (ESANN), pp. 81–94, Université catholique de Louvain, Brugge 2001.

[3] ANDREWS R., DIEDERICH J. TICKLE A.B. Survey and critique of techniques for extracting rules from trained artificial neural networks, Knowledge-Based Systems, Vol. 8, No. 6, pp. 373–389, Queensland, 1995.

[4] CRAVEN M., SHALVIK J., Using neural networks for data mining, Future Generation Computer Systems, Vol. 13, pp. 211–229, Carnegie Mellon, 1997.

[5] KASKI S., Data exploration using self-organizing maps, Acta Polytechnica Scandinavica, Mathematics, Computing and Management in Engineering Series, No. 82, pp. 57, Helsinki, 1997.

[6] SETINO R., LOEW W.K., FERNN: An algorithm for fast extraction of rules from neural networks, Applied Intelligence, Vol. 12, pp. 15-25, Shrewsbury, 2000.

[7] COHEN S., INTRATOR N., A hybrid projection based and radial basis function architecture: initial values and global optimization, Pattern Analysis & Applications, Vol. 5, No. 2, pp. 113–120, Springer-Verlag, London, 2002.

[8] BLANZIERI E., Theoretical interpretations and applications of radial basis function networks, Technical Report DIT-03-023, Informatica e Telecomunicazioni, University of Trento, 2003.

[9] MHASKAR H.N., MICCHELLI C. A., Approximation by superposition of sigmoidal and radial basis functions, Advances in Applied Mathematics, Vol. 13, No. 3, pp. 350–373, Orlando, 1992.

[10] CHEN T., CHEN R., Universal approximation to nonlinear operators by neural networks with arbitrary activation functions and its application to dynamical systems, IEEE Transactions on Neural Networks, Vol. 6, No. 4, pp. 911–917, IEEE Press, 1995.

[11] KAVZOGLU T., MATHER P.M., The use of feature selection techniques in the context of artificial neural networks, Proc. 26th Annual Conference of the Remote Sensing Society (CD-ROM), University of Leicester, Leicester, 2000.

[12] FREITAS A.A. On rule interestingness measures, Knowledge-Based Systems, Vol. 12, No. 5, pp. 309–315, Elsevier Science, British Computer Society, 1999.

[13] FREITAS A.A. Understanding the crucial role of attribute interaction in data mining, Artificial Intelligence Review, Vol. 16, No. 3, pp. 177–199, Kluwer Academic Publishers, Norwell, 2001.

[14] HUK M., Określanie istotności atrybutów w zadaniach klasyfikacyjnych przez niedestruktywną eliminację połączeń w sieci neuronowej, Pozyskiwanie Wiedzy z Baz Danych, Vol. 975, pp. 138–147, Akademia Ekonomiczna, Wrocław, 2003.

[15] PRECHELT L., Connection pruning with static and adaptive schedules, Neurocomputing, Vol. 16, No. 1, pp. 49–61, Elsevier Science, Karlsruhe, 1997.

[16] PUI-FAI SUM J., Extended Kalman filter based pruning algorithms and several aspects of neural network learning, PhD Dissertation, Department of Computer Science and Engineering, Chinese University of Hong Kong, 1998.

[17] ABELES M., Role of the cortical neuron: integrator or coincidence detector ?, Israel Journal of Medical Science, Vol. 18, No. 1, pp. 83–92, Weizmann Science Press, Jerusalem, 1982.

[18] MAASS W., Bishop C., Pulsed neural networks, pp. 261–293, MIT Press, Cambridge, 1999.

[19] MAASS W., Paradigms for computing with spiking neurons, In J. L. van Hemmen, J. D. Cowan, and E. Domany, editors, Models of Neural Networks, Early Vision and Attention, Vol. 4, chapter 9, pp. 373–402, Springer-Verlag, New York, 2002.

[20] BOTHE S., LA POUTRE J., KOK J., Error-backpropagation in temporally encoded networks of spiking neurons, CWI Technical Report SEN-R0036, Stichting Mathematisch Centrum, Amsterdam, 2000.

[21] MANGASARIAN O.L., SOLODOV M.V., Serial and parallel backpropagation convergence via nonmonotone perturbed minimization, Optimization Methods and Software, Vol. 4, pp. 103–116, Taylor & Francis Group, Stanford, 1994.

[22] LUO Z.A., TSENG P., Analysis of an approximate gradient projection method with application to the backpropagation algorithm, Optimization Methods and Software, Vol. 4, pp. 85–102, Taylor & Francis Group, Stanford, 1994.

[23] OHNO-MACHADO L., MUSEN M.A., Modular neural networks for medical prognosis: quantifying the benefits of combining neural networks for survival prediction, Connection Science, Vol. 9, No. 1, pp. 71–86, Taylor & Francis Group, Stanford, 1995.

[24] LIU Y., YAO X., Neural networks for breast cancer diagnosis, Proc. 1999 Congress on Evolutionary Computation, pp. 1760–1767, IEEE Press, New York, 1999.

[25] OSSEN A., ZAMZOW T., Segmentation of medical images using neural-network classifiers, Proc. First International Conference on Neural Networks and Expert Systems in Medicine and Healthcare, pp. 472–432, Plymouth, 1994.

[26] ANDERSON C., CRAWFORD-HINES S., Learning expert delineations in biomedical image segmentation, Proc. Conference on Artificial Neural Networks In Engineering, pp. 657–662, St. Louis, 2000.

[27] VEROPOULOS K., CAMPBELL C., SIMPSON J., The automated identification of tubercle bacilli using image processing and neural computing techniques, Proc. International Conference on Artificial Neural Networks, Vol. 2, pp. 797–802, Springer, Skövde, 1998.

[28] VEROPOULOS K., Machine learning approaches to medical decision making, PhD Thesis, Department of Computer Science, University of Bristol, March 2001.

[29] CIOS K.J., MOORE G.W., Uniqueness of medical data mining, Artifitial Intelligence in Medicine, Vol. 26, No. 1, pp. 1–24, Elsevier Science, 2002.

[30] GAMBERGER D., LAVRAC N., KRSTATIC G., SMUC T., Inconsistency tests for patients records in a coronary heart disease database, Proc. First International Symposium on Medical Data Analysis, pp. 183–189, Springer, Frankfurt, 2000.

[31] GAMBERGER D., LAVRAC N., GROSELI C., Experiments with noise filtering in a medical domain., Proc. 16th International Conference on Machine Learning, pp. 143–151, Jožef Stefan Institute, Bled, 1999.

[32] KUKAR M., KONONENKO I., GROSELJ C., KRALJ K., FETTICH J., Analysing and improving the diagnosis of ischaemic heart disease with machine learning, Artificial Intelligence in Medicine, Vol 16, No. 1, pp. 25–50, Elsevier Science, 1999.

[33] SHAHAR Y., CHENG C., Knowledge-based visualization of time-oriented clinical data. Proc. American Medical Informatics Association Fall Symposium, pp. 155-159, Orlando, 1998.

The above research in the area of intelligent information systems and machine learning is funded by the Polish Ministry of Scientific Research and Information Technology within the 4 T11C 02425 research grant.