UDC 004.272.26: 004.93

ADDITIONAL TRAINING OF NEURO-FUZZY DIAGNOSTIC MODELS

Oliinyk A. – PhD, Associate Professor, Associate Professor of the Department of Software Tools, Zaporizhzhia National Technical University, Zaporizhzhia, Ukraine.

Subbotin S. – Dr. Sc., Professor, Head of the Department of Software Tools, Zaporizhzhia National Technical University, Zaporizhzhia, Ukraine.

Leoshchenko S. – student of the Department of Software Tools, Zaporizhzhia National Technical University, Zaporizhzhia, Ukraine.

Ilyashenko M. – PhD, Associate Professor, Associate Professor of the Computer Systems and Networks Department, Zaporizhzhia National Technical University, Zaporizhzhia, Ukraine.

Myronova N. – PhD, Associate Professor of the Department of Software Tools, Zaporizhzhia National Technical University, Zaporizhzhia, Ukraine.

Mastinovsky Y. – PhD, Professor, Head of the Department of Applied Mathematics, Zaporizhzhia National Technical University, Zaporizhzhia, Ukraine.

ABSTRACT

Context. The task of automation of diagnostic models synthesys in diagnostics and pattern recognition problems is solved. The object of the research are the methods of the neuro-fuzzy diagnostic models synthesys. The subject of the research are the methods of additional training of neuro-fuzzy networks.

Objective. The research objective is to create a method for additional training of neuro-fuzzy diagnostic models.

Method. The method of additional training of diagnostic neuro-fuzzy models is proposed. It allows to adapt existing models to the change in the functioning environment by modifying them taking into account the information obtained as a result of new observations. This method assumes the stages of extraction and grouping the correcting instances, diagnosing them with the help of the existing model leads to incorrect results, as well as the construction of a correcting block that summarizes the data of the correcting instances and its implementation into an already existing model. Using the proposed method of learning the diagnostic neural-fuzzy models allows not to perform the resource-intensive process of re-constructing the diagnostic model on the basis of a complete set of data, to use the already existing model as the computing unit of the new model. Models synthesized using the proposed method are highly interpretive, since each block generalizes information about its data set and uses neuro-fuzzy models as a basis.

Results. The software which implements the proposed method of additional training of neuro-fuzzy networks and allows to reconfigure the existing diagnostic models based on new information about the researched objects or processes based on the new data has been developed.

Conclusions. The conducted experiments have confirmed operability of the proposed method of additional training of neuro-fuzzy networks and allow to recommend it for processing of data sets for diagnosis and pattern recognition in practice. The prospects for further researches may include the development of the new methods for the additional training of deep learning neural networks for the big data processing.

KEYWORDS: data sample, diagnosis, additional training, neuro-fuzzy model, parameter, membership function.

ABBREVIATIONS

BPSad is a back propagation for additional training based on sample S_{ad};

BPS is a back propagation for re-training based on sample S;

MATDNFM is a method for additional training of the diagnostic neuro-fuzzy models;

NFN is a neuro-fuzzy network.

NOMENCLATURE

b_{mj} is a parameter of the membership function;

$|CS'|$ is a number of instances cs'_q of the set CS';

cs'_{mq} is a m-th coordinate of the q-th instance $cs'_q \in CS'$;

C_{mj} is a m-th coordinate of the center of the j-th cluster C_j;

d_{mj} is a parameter of the membership function;

$\varepsilon_{\min}$ is a minimum acceptable difference between the real and model values of the output parameter;

ε_J is a minimum acceptable change in the value of the criterion J;

M is a number of features in the sample of observations S;

$\mu_{\min}$ is a parameter that determines the minimum acceptable membership degree of the instance s_{new} to the set $S' = <P', T'>$ of data on the basis of which the correcting block NB was synthesized;

N_R is a number of model NFN rules;

P is a set of features (attributes) of observations in the given sample;

p_{qm} is a value of the m-th feature (attribute) of the q-th observation;

$p_m(s_{new})$ is a m-th coordinate of the evaluated instance $s_{new} \notin S$;

Q is a number of observations in the given sample of observations S;

$\text{Round}(a)$ is a function that returns the result of rounding the number a to the nearest larger integer;

S is a sample of observations (training sample);

DOI 10.15588/1607-3274-2018-3-12

t_q is a value of output parameter of the q-th observation;

t'_q is a measured value of the output parameter of the q-th instance s'_q of sample $S' =< P',T' >$;

$t'_q(NFN)$ is a value of the output parameter of the q-th instance s'_q of sample $S' =< P',T' >$, calculated by substituting the measured values of the input attributes p'_{qm} of the q-th instance in the model NFN;

T is a set of output parameter values;

$t_{q\,\mathrm{mod}}$ is a model value of the output parameter of the q-th instance cs'_q, calculated from the synthesized model y_{NBj};

u_{qj} is a value of membership function of the q-th instance cs'_q to the j-th cluster;

w_{mj} is a customizable parameter of the function y_{NBj}.

INTRODUCTION

During operation of intelligent diagnostic systems, new information about diagnosed objects arises. In doing so, the information newly obtained from the measurements of the diagnosed objects can significantly contradict to the existing diagnostic models built on the results of previous observations. In such cases, it becomes necessary to re-synthesize diagnostic models using the data from previous and new measurements.

The object of study are the methods of the neuro-fuzzy diagnostic models synthesys.

However, when working with big data, the time for re-synthesis of such models can be significant, which in some cases is unacceptable. Therefore, during the operation of diagnostic systems, the task of adapting trained models by modifying them, taking into account the information obtained as a result of new observations, is relevant.

The subject of study are the methods of additional training of neuro-fuzzy networks.

The purpose of the work is to create a method for additional training of neuro-fuzzy diagnostic models.

1 PROBLEM STATEMENT

Suppose we have:

1) a sample of data $S =< P,T >$, containing Q instances, each of which is characterized by the values of the parameters p_{q1}, p_{q2}, …, p_{qM} and the output parameter t_q;

2) a neuro-fuzzy model $NFN = NFN(struct, param)$ synthesized from a set of observations $S =< P,T >$ with a definite structure *struct* (a set of computational elements connected in a certain way) and set of parameters $param = param(struct)$;

3) a data set $S' =< P',T' >$ obtained as a result of new Q' measurements of the object being examined (diagnosed).

Then it is necessary to synthesize the new model $NFNN = NFNN(structN, paramN)$ by modifying the existing model $NFN(struct, param)$ taking into account the new data $S' =< P',T' >$ in such a way that an acceptable value of the specified quality criterion G of the neuromodel $NFNN$: $G(NFNN, S \cup S') \to \min$ is provided. For example, a minimum of recognition error (in problems with a digital output T) or a minimum mean-square error (in the case where the output parameter T can take real values from a certain range $T \in [t_{\min};t_{\max}]$) can be used as the target criterion G for additional training neural-fuzzy models.

2 REVIEW OF THE LITERATURE

The additional training of diagnostic and recognition models built in the form of neural-fuzzy networks usually involves the modification of the existing network by including (adding) information about new observations to it. Such information is added to the constructed network in the form of new rules, represented by so-called singletons. This approach is simple enough to implement. However, in the case of a significant number of new observations, the application of this approach is little effective. The reason is that in this case the structural and parametric complexity of the network is significantly increased (each new observation, in fact, is added to the network in the form of a new rule), and its generalizing capabilities are also reduced.

Another approach involves a complete reorganization of the structure and parameters of the network with the appearance of new essential information about the objects under study. Consequently, the already synthesized model is re-trained on the basis of available $S =< P,T >$ and new $S' =< P',T' >$ information. When processing big data, re-training the model is also undesirable, since this process takes a lot of time and requires a large amount of computational resources.

Therefore, it is advisable to develop a new method for adapting trained neural-fuzzy models to changing the functioning environment by modifying them, taking into account the information obtained as a result of new observations.

3 MATERIALS AND METHODS

In the developed method of training the neuro-fuzzy models, it is proposed to correct the existing model $NFN(struct, param)$ by introducing additional structural computational elements that take into account the attributes of the new data set $S' =< P',T' >$.

DOI 10.15588/1607-3274-2018-3-12

In the proposed method, the first step is to extract the correcting instances from the sample $S' = <P', T'>$. Corrective instances cs'_q will be considered those observations of the sample $S' = <P', T'>$, diagnosing them using the existing model $NFN(struct, param)$ leads to incorrect results. Consequently, the diagnosing model NFN used needs to be adjusted precisely with the help of instances cs'_q.

Therefore, to construct a set of corrective instances CS', all sample $S' = <P', T'>$ instances s'_q are passed through the model NFN, as a result of which the value of the output parameter $t'_q(NFN)$ of each q-th instance of sample $S' = <P', T'>$ is calculated. Then, the real t'_q and model $t'_q(NFN)$ values of the output parameter are compared:

$$\left|t'_q - t'_q(NFN)\right| \geq \varepsilon_{\min}. \quad (1)$$

Condition (1) is used in solving estimation problems (for continuous values of the output parameter *T*). When solving recognition problems (with discrete values of the output parameter *T*) the following condition is used: $t'_q \neq t'_q(NFN)$.

When the above conditions are met, the q-th instance s'_q of sample $S' = <P', T'>$ is counted as corrective and entered into the set CS': $CS' = CS' \bigcup s'_q$. Thus, as a result of the step of extracting the correcting instances, those instances of sample $S' = <P', T'>$ that are similar to the instances of the original sample $S = <P, T>$ are excluded from further consideration and, therefore, do not affect the quality of recognition or estimation by model NFN.

Later, instances cs'_q of set CS' can be used as singletons in constructing a new block $NB(structNB, paramNB)$, introduced along with the already existing model $NFN(struct, param)$ in the new model $NFNN = NFNN(structN, paramN)$.

However, when processing big data, the number of set`s CS' instances can be significant, which will lead to a significant increase in the structural and parametric complexity of the new model $NFNN$. In addition, many instances of the set CS' can be close to each other in the attribute space and, in fact, be similar. Therefore, including all instances $cs'_q \in CS'$ as rules for a new model block can also lead to a loss of its generalizing abilities.

Accordingly, before building a block NB, it is advisable to perform the step of grouping the correcting instances of the set CS' with the selection of the most significant of them $csInf'_q$, concentrating around themselves a certain number of similar closely located specimens.

To do this, it is suggested to perform cluster analysis of the CS' set`s instances in the attribute space P. The number of clusters N_{Cl} in the developed method is determined in proportion to the number of rules N_R in the existing model NFN, as well as the proportion of instances $|CS'|$ of the set CS' in relation to the number of instances Q in the set $S = <P, T>$ (2):

$$N_{Cl} = \text{Round}\left(\frac{|CS'|}{Q} N_R\right). \quad (2)$$

After determining the number of clusters N_{Cl}, the initial partitioning of instances $cs'_q \in CS'$ over clusters is performed. For this, a set of cluster centers $C = \{C_1, C_2, ..., C_{NCl}\}$ is defined, where $C_j = \{C_{1j}, C_{2j}, ..., C_{Mj}\}$ is the center of the j-th cluster, $j = 1, 2, ..., N_{Cl}$. The centers C_j can be selected randomly among instances cs'_q of the set CS'. It is also possible to create a set $C = \{C_1, C_2, ..., C_{NCl}\}$ taking into account the spatial arrangement of the instances $cs'_q \in CS'$. For this, an instance cs'_a is first randomly selected from CS', which is considered the center of the first cluster $C_1 = \{cs'_{1a}, cs'_{2a}, ..., cs'_{Ma}\}$. Then, as the center of the second cluster C_2, the instance cs'_b most remote from the instance cs'_a is selected. The center of the third cluster C_3 is selected in such a way that it is as far away from the centers of the first and second clusters. This procedure continues until N_{Cl} is formed. With a large value N_{Cl}, this approach will be associated with the need for complex calculations due to the search for instances characterized by the greatest distance to the current set of already defined cluster centers. Therefore, this approach is advisable to apply for small values of the number of clusters N_{Cl} or to combine it with an approach that involves the random formation of multiple cluster centers $C = \{C_1, C_2, ..., C_{NCl}\}$.

Then, the generation of elements u_{qj} determining the membership of the q-th instance cs'_q to the j-th cluster C_j is performed. In contrast to the method of fuzzy c-means used as a basis, in the developed method, when creating the initial division of the instances, the generation of elements u_{qj} will be performed not randomly, but taking into account the location of the instances $cs'_q \in CS'$ in attribute space P. For this, the distances $D(cs'_q, C_j)$ from the instance cs'_q to the center C_j of each cluster $j = 1, 2, ..., N_{Cl}$ are determined. As a metric for determining the distance $D(cs'_q, C_j)$, we can use the Euclidean metric (3):

DOI 10.15588/1607-3274-2018-3-12

$$D\left(cs'_q, C_j\right) = \sqrt{\sum_{m=1}^{M}\left(cs'_{mq} - C_{mj}\right)^2}\ . \qquad (3)$$

The membership u_{qj} of the *q*-th instance cs'_q to the *j*-th cluster C_j is calculated by the formula (4):

$$u_{qj} = \left(\sum_{JA=1}^{N_{Cl}}\left(\frac{D\left(cs'_q, C_j\right)}{D\left(cs'_q, C_{JA}\right)}\right)^{\frac{2}{mp-1}}\right)^{-1}. \qquad (4)$$

In the case where the instance cs'_q is the center of the *j*-th cluster C_j ($D\left(cs'_q, C_j\right) = 0$), then it is established: $u_{qj} = 1$, $u_{qJA} = 0, \forall JA \neq j$.

Further, according to the formula (5), the value of the function $J\left(R^{(i)}, u^{(i)}, C^{(i)}\right)$ determining the quality of the fuzzy partitioning $R^{(i)}$ in the *i*-th iteration of the cluster analysis is calculated:

$$J\left(R^{(i)}, u^{(i)}, C^{(i)}\right) = \sum_{q=1}^{|CS'|}\sum_{j=1}^{N_{Cl}}\left(u_{qj}\right)^{mp} D^2\left(cs'_q, C_j\right). \qquad (5)$$

After that, the criteria (6) and (7) of the completion of the cluster analysis procedure are checked:

$$\left|J_{old}\left(R^{(i)}, u^{(i)}, C^{(i)}\right) - J\left(R^{(i)}, u^{(i)}, C^{(i)}\right)\right| \leq \varepsilon_J\ , \qquad (6)$$

$$i \geq maxIterClA. \qquad (7)$$

In this case, inequality (6) reflects a condition, the fulfillment of which characterizes too small a change in the value of the target function $J\left(R^{(i)}, u^{(i)}, C^{(i)}\right)$, and accordingly, the inexpediency of further searching for the optimal partition $R^{(i)}$. Condition (7) displays the situation when the current number of iterations reaches the maximum allowed value $maxIterClA$. If both conditions (6) and (7) are not fulfilled, the new values of the coordinates of the cluster centers are determined using formula (8):

$$C_{mj} = \frac{\sum_{q=1}^{|CS'|}\left(u_{qj}\right)^{mp} cs'_{mq}}{\sum_{q=1}^{|CS'|}\left(u_{qj}\right)^{mp}}. \qquad (8)$$

Then, using the formulas (3)–(5), a new fuzzy partition $R^{(i+1)}$ is searched (allocation of accessories u_{qj}). This procedure is repeated until at least one of the conditions (6) or (7) is satisfied.

Consequently, as a result of the step of grouping the correcting instances, a plurality of cluster centers $C = \{C_1, C_2, ..., C_{NCl}\}$ and a plurality of cs'_q instance attachments u_{qj} are formed to the respective clusters.

After grouping the correcting instances, the stage of construction of the correcting block NB is performed.

In case the modifiable model $NFN(struct, param)$ uses as a basis a neuro-fuzzy ANFIS network, then the structure of the correction block NB will also be based on the ANFIS network. The graphic representation of the correcting block NB is shown in Fig. 1.

In this case, the number N_{RNB} of nodes of the second layer corresponding to fuzzy rules in the correcting block NB is proposed to be taken equal to the number of clusters (rules) allocated in the previous step: $N_{RNB} = N_{Cl}$. Given the nature of calculating parameter N_{Cl} in the proposed method, the number of NB-block rules N_{RNB} will be proportional to the number of rules N_R in the existing model NFN, as well as the proportion of instances $|CS'|$ of the set CS' in relation to the number of instances Q in the set $S = < P, T >$. Therefore, the structural complexity of the correcting block N_{RNB} will be proportional to the analogous value of the original model NFN and the proportion of new instances of the CS' set.

Neural elements of the first layer that determine the membership degree of the value of the input parameter p_m to the corresponding fuzzy term ft_{mj} ($j = 1, 2, ..., N_{RNB}$) are connected with the corresponding nodes of the second layer. Thus, in aggregate, the nodes of the first and second layers form antecedents of fuzzy rules NR_j.

The information obtained at the previous stages of the developed method of additional training the neural-fuzzy models (a multiplicity of correcting instances CS', a multiplicity of cluster centers $C = \{C_1, C_2, ..., C_{NCl}\}$ and a multiplicity of instance cs'_q accessories u_{qj} to the corresponding clusters) will be used to determine the configurable parameters of membership functions $\mu^{(1)}_{NBmj}$ ($m = 1, 2, ..., M$, $j = 1, 2, ..., N_{RNB}$). As functions $\mu^{(1)}_{NBmj}$ that determine the membership degree of the value of the m-th input parameter p_m to the j-th fuzzy term ft_{mj} in the correcting block NB, we use the membership functions (9):

$$\mu^{(1)}_{NBmj}\left(p_m\right) = \exp\left(-\frac{\left(p_m - b_{mj}\right)^2}{2{d_{mj}}^2}\right). \qquad (9)$$

As a parameter b_{mj} ($m = 1, 2, ..., M$, $j = 1, 2, ..., N_{RNB}$), that determines the shift of the center of the function relative to the center of coordinates of the characteristic axis p_m, we will use the m-th coordinate of the *j*-th cluster center C_j from the set $C = \{C_1, C_2, ..., C_{NCl}\}$ formed in the previous step.

DOI 10.15588/1607-3274-2018-3-12

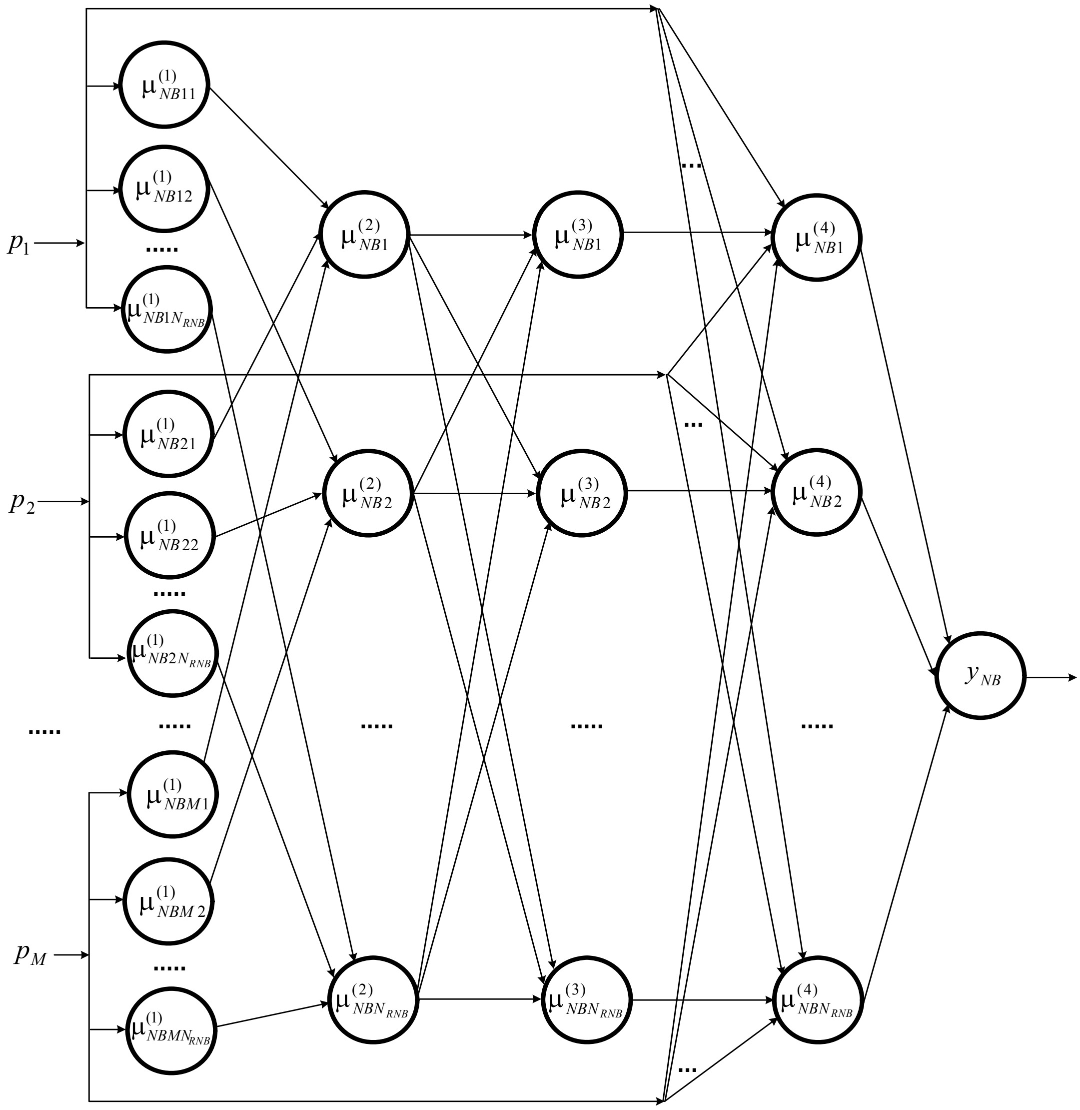

Figure 1 – Graphical interpretation of the correcting block NB when modifying models NFN using as a basis a neural-fuzzy ANFIS network

As a parameter d_{mj}, we will use the standard deviation of the correcting instances $cs'_q \in CS'$ relative to j-th center of the cluster C_j along the m-th characteristic axis. It also takes into account the membership u_{qj} of q-th correcting instance $cs'_q \in CS'$ to j-th cluster C_j:

$$d_{mj} = \sqrt{\frac{1}{|CS'|}\sum_{q=1}^{CS'} u_{qj}\left(cs'_{mq} - C_{mj}\right)^2} \,. \quad (10)$$

Using formulas (10) and (11) to define custom parameters b_{mj} and d_{mj} membership functions $\mu^{(1)}_{NBmj}$, in the process of evaluating new instances $s_{new} \notin S$ using the correcting block NB, activate those fuzzy terms ft_{mj} that together ($m = 1, 2, ..., M$) correspond to certain clusters C_j (NR_j fuzzy rules).

Having determined $\mu^{(1)}_{NBmj}$ with account of the calculated estimates b_{mj}, d_{mj}, it is possible to calculate the values of the outputs of the second layer of the network $\mu^{(2)}_{NBj}$ that determine the degree of fulfillment of the j-th rule NR_j, according to the formula:

$$\mu^{(2)}_{NBj}(s_{new}) = \bigcap_{m=1}^{M} \mu^{(1)}_{NBmj}\left(p_m(s_{new})\right). \quad (11)$$

The nodes of the third layer determine the relative degree of fulfillment of the j-th rule NR_j:

$$\mu^{(3)}_{NBj}(s_{new}) = \frac{\mu^{(2)}_{NBj}(s_{new})}{\sum_{JB=1}^{N_{RNB}} \mu^{(2)}_{NBJB}(s_{new})} \,. \quad (12)$$

Neural elements of the fourth layer $\mu^{(4)}_{NBj}$ ($j = 1, 2, ..., N_{RNB}$) correspond to functions y_{NBj} that

DOI 10.15588/1607-3274-2018-3-12

determine the value of the network output in the case of the operation of the corresponding rule NR_j. Thus, each j-th node of the network determines the contribution of the fuzzy rule NR_j to the common output of the network y_{NB}. Functions y_{NBj}, as a rule, are represented in the form of a linear regression, therefore, the values of the outputs of the nodes of the fourth layer $\mu_{NBj}^{(4)}$ can be calculated from the formula (13):

$$\mu_{NBj}^{(4)}(s_{new}) = \mu_{NBj}^{(3)}(s_{new}) y_{NBj}(s_{new}) = \\ = \mu_{NBj}^{(3)}(s_{new}) \left(\sum_{m=0}^{M} w_{mj}(p_m(s_{new})) \right). \quad (13)$$

It is assumed that $p_0(s_{new}) = 1$, and w_{0j} coefficient corresponds to the value of the free linear regression term (13). The function y_{NBj} can be simplified as follows: $y_{NBj} = \sum_{m=0}^{M} w_{mj} p_m$. More complex nonlinear dependencies can also be used as a basis of functions y_{NBj}.

Therefore, in order to synthesize a correcting block NB, it is necessary to restore the functions y_{NBj}, having determined the values w_{mj} of the adjustable parameters for this.

To determine the values of parameters w_{mj} in the developed method, it is proposed to use information not only about the values of the coordinates of the correcting instances $cs'_q \in CS'$, but also information on their membership degree u_{qj} to each of the clusters Cl_j (in fact, the fuzzy rule NR_j) determined by the centers $C = \{C_1, C_2, ..., C_{NCl}\}$. This will take into account the importance of the instances $cs'_q \in CS'$ for restoring the functions y_{NBj} corresponding to clusters Cl_j, and in determining the w_{mj} parameters of the function y_{NBj}, increase the contribution of those specimens that are characterized by high estimates of the membership degree of u_{qj} to the cluster Cl_j.

There are two approaches to determine the parameter w_{mj} values.

The first approach involves building models y_{NBj} on the basis of corrective instances cs'_q with the maximum estimates u_{qj} of membership to the corresponding clusters Cl_j. For each cluster Cl_j (NR_j rule), the instances cs'_q with the largest values of u_{qj} are selected. So, it is considered that the instance cs'_q belongs to the cluster Cl_j ($cs'_q \in Cl_j$) when the condition $u_{qj} = \max_q (u_{qj})$ is met. If this condition is met, then the instance cs'_q is added to the set Set_j of instances related to the cluster Cl_j: $Set_j = Set_j \cup cs'_q$. Further, using instances of the set Set_j, the function y_{NBj} is restored using the known parametric synthesis of models.

The second approach involves the use of all instances cs'_q of the set CS' to construct all models y_{NBj}. When restoring the function y_{NBj} that determines the output of j-th node of the fourth layer of the correcting block NB, all instances $cs'_q \in CS'$ are used, and also the membership u_{qj} of each of them to j-th cluster Cl_j (rule NR_j) is taken into account.

In the process of restoring the function y_{NBj} as the objective function E, we will use the function (14), which is a modified mean-square error function:

$$E_j = \frac{1}{|CS'|} \sum_{q=1}^{|CS'|} u_{qj} (t_q - t_{q\,\mathrm{mod}})^2. \quad (14)$$

The model value $t_{q\,\mathrm{mod}}$ of the output parameter of the q-th instance cs'_q is calculated from the synthesized model y_{NBj}:

$$t_{q\,\mathrm{mod}} = y_{NBj}(cs'_q) = \sum_{m=0}^{M} w_{mj}(p_m(cs'_q)). \quad (15)$$

This function, in addition to the deviation between the actual t_q and model value $t_{q\,\mathrm{mod}}$ of the output parameter of the instances cs'_q, also uses information about the membership u_{qj} of this instance to j-th cluster Cl_j as an estimate of the importance of the instance cs'_q for restoring the function y_{NBj}.

Substituting (15) into (14), reducing obtained expression by a multiplier $\frac{1}{|CS'|}$ and taking into account that $p_m(cs'_q) = p_{qm}$, we obtain the objective function of the form (16):

$$E_j = \sum_{q=1}^{|CS'|} u_{qj} \left(t_q - \sum_{m=0}^{M} w_{mj} p_{qm} \right)^2 = \\ = \sum_{q=1}^{|CS'|} \left(u_{qj}(t_q)^2 - 2u_{qj} t_q \sum_{m=0}^{M} w_{mj} p_{qm} + u_{qj} \left(\sum_{m=0}^{M} w_{mj} p_{qm} \right)^2 \right). \quad (16)$$

To determine the values of adjustable parameters w_{mj}, find their values at which optimum target criterion $E_j \to opt$ is reached. To do this, define the partial derivatives by the parameters w_{mj} of the target criterion E_j as functions of several variables: $E_j = E_j(w_{0j}, w_{1j}, w_{Mj})$, then solve the system of equations (17):

DOI 10.15588/1607-3274-2018-3-12

$$\frac{\partial E_j}{\partial w_{mj}} = 0, \quad m = 0,1,2,...,M. \tag{17}$$

Expression (17) is a system consisting of $(M+1)$ linear equations of the form (18)

$$\frac{\partial E_j}{\partial w_{mj}} = \sum_{q=1}^{|CS'|}\left(-2u_{qj}t_q p_{qm} + 2u_{qj}p_{qm}\left(\sum_{m=0}^{M} w_{mj}p_{qm}\right)\right) = 0 \tag{18}$$

or

$$\sum_{q=1}^{|CS'|}\left(u_{qj}p_{qm}\left(w_{0j}p_{q0} + w_{1j}p_{q1} + ... + w_{Mj}p_{qM}\right)\right) = \sum_{q=1}^{|CS'|}\left(u_{qj}t_q p_{qm}\right), \quad m = 0,1,2,...,M.$$

Performing further transformations, obtain that the m-th equation of system (17) can be written in the form (19):

$$w_{0j}\sum_{q=1}^{|CS'|}\left(u_{qj}p_{qm}p_{q0}\right) + w_{1j}\sum_{q=1}^{|CS'|}\left(u_{qj}p_{qm}p_{q1}\right) + \\ + ... + w_{Mj}\sum_{q=1}^{|CS'|}\left(u_{qj}p_{qm}p_{qM}\right) = \sum_{q=1}^{|CS'|}\left(u_{qj}t_q p_{qm}\right). \tag{19}$$

Substituting the values $m = 0,1,2,...,M$ in (19), we obtain a system of linear algebraic equations (20):

$$\begin{cases} w_{0j}\sum_{q=1}^{|CS'|}\left(u_{qj}p_{q0}p_{q0}\right) + w_{1j}\sum_{q=1}^{|CS'|}\left(u_{qj}p_{q0}p_{q1}\right) + ... + w_{Mj}\sum_{q=1}^{|CS'|}\left(u_{qj}p_{q0}p_{qM}\right) = \sum_{q=1}^{|CS'|}\left(u_{qj}t_q p_{q0}\right); \\ w_{0j}\sum_{q=1}^{|CS'|}\left(u_{qj}p_{q1}p_{q0}\right) + w_{1j}\sum_{q=1}^{|CS'|}\left(u_{qj}p_{q1}p_{q1}\right) + ... + w_{Mj}\sum_{q=1}^{|CS'|}\left(u_{qj}p_{q1}p_{qM}\right) = \sum_{q=1}^{|CS'|}\left(u_{qj}t_q p_{q1}\right); \\ .... \\ w_{0j}\sum_{q=1}^{|CS'|}\left(u_{qj}p_{qM}p_{q0}\right) + w_{1j}\sum_{q=1}^{|CS'|}\left(u_{qj}p_{qM}p_{q1}\right) + ... + w_{Mj}\sum_{q=1}^{|CS'|}\left(u_{qj}p_{qM}p_{qM}\right) = \sum_{q=1}^{|CS'|}\left(u_{qj}t_q p_{qM}\right). \end{cases} \tag{20}$$

Further, solving the system (20) by the known methods of linear algebra, the required values w_{mj} are found.

As noted above, not only linear functions of the form ($y_{NBj} = \sum_{m=0}^{M} w_{mj}p_m$), but also more and more complex nonlinear dependencies can be used as functions y_{NBj}. In such cases, the search for configurable parameters w_{mj} of y_{NBj} function is performed by optimizing the target functional (14) using known gradient (in the case of functions y_{NBj} that are differentiable) or stochastic methods.

After determining the values of the adjustable parameters w_{mj} of the functions y_{NBj}, the values of the outputs $\mu^{(4)}_{NBj}$ of the neural elements of the fourth layer of the correcting block NB can be calculated from formula (13). Then, the total output of the block NB (21) is calculated:

$$y_{NB}(s_{new}) = \sum_{j=1}^{N_{RNB}} \mu^{(4)}_{NBj}(s_{new}). \tag{21}$$

Then, the stage of combining the existing model NFN and the correcting block NB is performed. At this stage, the information about the data sets $S = <P,T>$ and $S' = <P',T'>$, approximated by the model NFN and the correcting block NB, is generalized, respectively. The modified model $NFNN$ by adding the correcting block NB to the existing model $NFN(struct, param)$ is shown in Fig. 2.

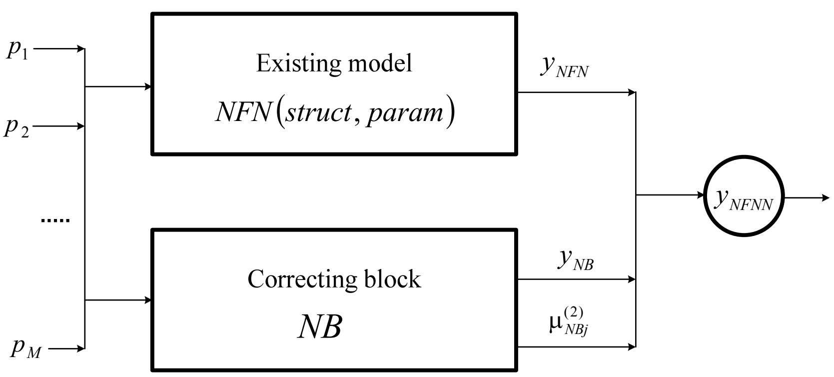

Figure 2 – Graphical interpretation of the modified model $NFNN$

DOI 10.15588/1607-3274-2018-3-12

e-ISSN 1607-3274 Радіоелектроніка, інформатика, управління. 2018. № 3
p-ISSN 2313-688X Radio Electronics, Computer Science, Control. 2018. № 3

To evaluate the value of the output parameter $t(s_{new})$ of a new instance $s_{new} \notin S$ using the model $NFNN$ modified (adapted to the new conditions) based on the new data $S' = < P', T' >$, it is proposed to use the formula (22):

$$t(s_{new}) = y_{NFNN}(s_{new}) = \begin{cases} y_{NB}(s_{new}), & \bigcup_{j=1}^{N_{RNB}} \mu_{NBj}^{(2)} > \mu_{\min}; \\ y_{NFN}(s_{new}), & \bigcup_{j=1}^{N_{RNB}} \mu_{NBj}^{(2)} \le \mu_{\min}. \end{cases} \quad (22)$$

As can be seen, if the new instance $s_{new} \notin S$ is characterized by a sufficiently large degree of belonging ($\bigcup_{j=1}^{N_{RNB}} \mu_{NBj}^{(2)} > \mu_{\min}$) to the rules NR_j of the correcting block NB (accordingly, to clusters Cl_j synthesized on the basis of a new data set $S' = < P', T' >$), the final value y_{NFNN} is calculated using the correcting block NB. Otherwise, it is considered that the instance s_{new} is more relevant to the source set $S = < P, T >$, and the value y_{NFNN} is taken equal to the output value y_{NFN} by the model NFN base.

It is important to note that the proposed approach to the construction of correcting blocks NB allows to synthesize and introduce into existing models new blocks with the appearance of new information $S' = < P', T' >$, the diagnosis of which leads to incorrect results of the model $NFNN$. Thus, the model shown in Fig. 2, can be consistently expanded by adding new blocks NB that generalize information about new observations of the investigated objects.

Consequently, the proposed method of additional training the neuro-fuzzy diagnostic models allows to adapt existing models to the change in the functioning environment by modifying them taking into account the information obtained as a result of new observations. The proposed method assumes the stages of extraction and grouping of correcting specimens, diagnosing with the help of the existing model leads to incorrect results, as well as the construction of a correcting block that summarizes the data of the correcting instances and its introduction into an already existing model. When determining the adjustable parameters of the correction block in the developed method, it is proposed to use information about the values of the coordinates of the correcting instances, as well as information on the degree of their membership to clusters in the feature space (and, accordingly, to the fuzzy rules presented in the correcting block). This allows one to take into account the importance of corrective instances for restoring the functions of the fourth layer of the correcting block and, when determining custom parameters, to increase the contribution of those specimens that are characterized by high estimates of membership degree to a particular cluster.

Using the proposed method of additional training the neural-fuzzy diagnostic models allows not to perform the resource-intensive process of re-constructing the diagnostic model on the basis of a complete set of data, to use the already existing model as the computing unit of the new model. In addition, models synthesized using the proposed method are highly interpretive, since each block generalizes information about its data set and uses neuro-fuzzy models as a basis.

4 EXPERIMENTS

For testing the effectiveness of the developed method training of neuro-fuzzy models training, the problem of constructing diagnostic models for predicting the health status of patients with hypertension was solved [30].

Hypertension is a widespread disease that can threaten the life and health of the patient [30]. The nature of the course of hypertension is influenced by various factors (weather and climatic conditions, concomitant diseases, as well as the state of health in previous moments) [30]. In order to prevent significant pressure surges that can cause deterioration of the patient's condition, and possibly lead to death, it is necessary to predict the development of hypertension in the short term (for the next half of the day or day). This will allow timely implementation of preventive measures related to the intake of necessary medicines to prevent the expected negative consequences.

For prediction the health of a patient with hypertension, it is necessary to have a model that will be unique for each individual patient. Building such a model requires processing a large array of observations distributed over time.

Thus, since such a disease is of an individual nature [30] (the features of the disease are different for each patient as a result of which for each patient it is necessary to synthesize its own unique diagnostic model) and in connection with obtaining new information about the course of the disease over time, there is a need for periodic adjustment (additional training) of existing models for individual prediction of the patient's condition on the basis of constantly growing arrays of observations.

The initial sample of data on the state of health of a patient with hypertension was obtained in Zaporizhzhia (Ukraine). The sample $S = < P, T >$ included observations from 2004 to 2014, where each sample was a set of data characterizing the patient's condition at a certain part of the day.

As objective clinical and laboratory features were used: p_1 is observed blood pressure (systolic and diastolic, mmHg); p_2 is a pulse (beats per minute (BPM)); data on medication (p_3 is an Amlo (0 is for patient that do not take medicines, 1 is for patient take medicines), p_4 is an Egilok (0 is for patient that do not take medicines, 1 is for patient take medicines); p_5 is a Berlipril (0 is for patient that do not take medicines, 1 is for patient

DOI 10.15588/1607-3274-2018-3-12

take medicines)). As subjective features used characteristics of health (p_6 is the presence of premature heart beat (0 is present, 1 is absent), p_7 is the presence of headache (0 is present, 1 is absent), p_8 is the presence of neck pain (0 is present, 1 is absent), p_9 is the presence of pulsation (0 is present, 1 is absent), p_{10} is the presence of pain in the left side (0 is present, 1 is absent), p_{11} is presence of pain in the heart (0 is present, 1 is absent), p_{12} is lack of air (0 is present, 1 is absent), p_{13} is presence of stomachache (0 is present, 1 is absent), p_{14} is general weakness (0 is present, 1 is absent)). As meteorological characteristics used [30] (p_{15} is an air temperature ($°C$)), p_{16} is an atmospheric pressure (mmHg), p_{17} is type of cloud cover (0 is not cloudy, 1 is small cloudy, 2 is cloudy, 3 is overcast), p_{18} is the presence of thunderstorms (0 is present, 1 is absent), p_{19} is wind direction (0 is a windless, 1 is a northern wind, 2 is a northeasterly wind, 3 is a easterly wind, 4 is a southeasterly wind, 5 is a southern wind, 6 is a southwesterly wind, 7 is a westerly wind, 8 is a northwesterly wind), p_{20} is a wind speed (m/s), p_{21} is a solar phenomena data (Mg II index). As characteristics of time were used: date (year, month, day), identification of the day of week (p_{22}), time (hour) of observation (p_{23}), identification of the part of day (0 is a morning, 1 is an evening) (p_{24}).

The observations obtained by the method of "Short-time transform" were used to form a sample to solve the problem of qualitative forecasting of the patient's condition for the next second half of the day according to the previous observations: as input features were used data for the previous (morning and evening) and the current day (morning), and as an output – the patient's condition in the evening in the current day (0 – normal, 1 – aggravation of symptoms, accompanied by an increase in blood pressure).

To carry out experiments on the researching of the developed method additional training of neuro-fuzzy diagnostic models the training sample $S=<P,T>$ was divided into two parts, the first S_{tr} of which was used for training (synthesis) of the model $NFN = NFN(struct, param)$, and the second S_{ad} – for additional training of the already synthesized model NFN in order to obtain a new model $NFNN = NFNN(structN, paramN)$. It is worth noting that the following conditions were met when splitting the sample $S=<P,T>$: $S_{tr} \bigcup S_{ad} = S$ and $S_{tr} \bigcap S_{ad} = \varnothing$.

Let the variable ω represents a relationship (23) of the cardinality of the sets S_{ad} and S_{tr}:

$$\omega = \frac{|S_{ad}|}{|S_{tr}|}. \quad (23)$$

The higher the value of the variable ω, the more new observations appeared after the previous construction (reconstruction) of the model NFN.

In the process of experimental studies will be applying different methods and approaches to the training of the constructed models at different values of the variable ω:

– additional training of the synthesized neuro-fuzzy model NFN using sample S_{ad} by the Backpropagation method (BPSad) [1, 2]. In this case, the parameters of the existing model NFN, pre-synthesized by sampling S_{tr}, were used as the initial parameters of the new model $NFNN$;

– re-training of the neuro-fuzzy model using the data of the combined sets $S_{tr} \bigcup S_{ad} = S$ (BPS);

– using the developed method for additional training for finishing the diagnostic neuro-fuzzy models (MATDNFM). Herewith, the finish of the model was performed on a sample S_{ad}, base model NFN was synthesized based on a sample S_{tr}.

As criteria for comparison the methods additional training of the neuro-fuzzy models shall be using:

– training time t_{ad} is an amount of time that was spent on building the model $NFNN$ (without taking into account the time that was used to synthesize the base model NFN);

– error E_S of the model $NFNN$ on the sample data $S=<P,T>$;

– error E_{Str} of the model on the sample data ;

– error E_{Sad} of the model on the sample data ;

– model error E_t on test data (observation data not reflected in the sample $S=<P,T>$).

5 RESULTS

The results of the experiments are given in table 1.

Table 1 – The results of experiments on the study of methods of neural-fuzzy networks training

ω, %	t_{ad}			E_S			E_{Str}			E_{Sad}			E_t		
	BPSad	BPS	MATDNFM	BPSad	BPS	MATDNFM	BPSad	BPS	MATDNFM	BPSad	BPS	MATDNFM	BPSad	BPS	MATDNFM
1	0.8139	82.632	0.6918	0.0936	0.0296	0.0296	0.0945	0.0299	0.0299	0	0	0	0.1756	0.0572	0.0555
10	7.4727	82.632	6.7255	0.0591	0.0296	0.0296	0.0649	0.0270	0.0324	0	0.0556	0	0.1109	0.0572	0.0555
20	13.7	82.632	11.645	0.0542	0.0296	0.0296	0.0592	0.0237	0.0237	0.0294	0.0588	0.0588	0.0637	0.0572	0.0555
50	27.4	82.632	20.824	0.0443	0.0296	0.0345	0.0519	0.0296	0.0296	0.0294	0.0294	0.0441	0.0521	0.0572	0.0647
100	41.1	82.632	25.893	0.0345	0.0296	0.0345	0.0294	0.0294	0.0294	0.0396	0.0297	0.0396	0.0406	0.0572	0.0647

DOI 10.15588/1607-3274-2018-3-12

6 DISCUSSION

Table 1 shows that the additional training time t_{ad}, that was spent on the construction of the model $NFNN$ using the method of BPS is constant ($t_{ad} = 82.632$ sec.) and does not depend on the value of the variable ω, because before training the neuro-fuzzy network with this approach is performed by using the entire data sample $S = < P, T >$ and does not depend on its division into the sample S_{tr}, which was used to train the basic model NFN, and the sample S_{ad} for the additional training of the already synthesized model NFN (building a new model $NFNN$. It should be noted that for small amounts of data (a low number of instances of the training sample $S = < P, T >$), the synthesis time of the model is acceptable. However, the use of this approach in the processing of BPS large data sets for the restructuring of the already synthesized models is undesirable, and in some cases impossible, because the process of learning (re-training) will require significant time and hardware resources of the computer.

The additional training time t_{ad} in the case using the method BPSad depends on the value of the variable ω (changes from 0.8139 sec. with $\omega = 1\%$ to 41.1 sec. with $\omega = 100\%$) due to the fact that a reduced sample S_{ad} is used as the sample for which the neuro-fuzzy model is being trained. Similar results shows the proposed method MATDNFM. However, the additional training time a few below (changes from 0.6918 c. with $\omega = 1\%$ to 25.893 c. with $\omega = 100\%$) compared to additional training time with using BPSad. This is conditioned by the fact that the proposed method is pre-grouping of new instances, thereby significantly reducing the number of new rules that are introduced in the neuro-fuzzy diagnostic model, and this, in turn, reduces the number of configurable parameters and, accordingly, the time of model learning.

The error E_S on the sample $S = < P, T >$ for the BPS method is constant ($E_S = 0.0296$) and such, which does not depend on the value of the variable ω, because the additional training (re-synthesis of the model) is performed throughout the data sample $S = S_{tr} \cup S_{ad}$. The error E_S for the BPSad method is slightly worse in comparison with the BPS (especially at low values of ω: $E_S = 0.0936$ with $\omega = 1\%$, $E_S = 0.0345$ with $\omega = 100\%$), because a reduced sample S_{ad} is used for additional training , which is only a certain part of the sample $S = < P, T >$. As can be seen from the table, the error E_S for the BPSad method decreases with increasing value of ω. This is conditioned by the increase in the sample S_{ad} share relative to $S = < P, T >$ the increase ω. It is worth noting that in solving practical problems the number of new data (value of ω) is usually significantly lower than the amount of initial information (sample $S = < P, T >$). This confirms the expediency of using the proposed method, in which the error E_S on the sample $S = < P, T >$ is almost unchanged (does not significantly depend) when the value of the variable ω change and is commensurate with the magnitude of the error E_S using the BPS approach. This is due to the use of formulas (22) to calculate the values of the output parameter T, which takes into account both the preliminary compilation of data samples S_{tr} in the form of the underlying model NFN (output parameter value $t(s)$ is calculated according to the basic model NFN in the case that the instance s has low degree of belonging $\bigcup_{j=1}^{N_{RNB}} \mu_{NBj}^{(2)} \le \mu_{\min}$ to the rules of the new structural element of the model in the form of the correction unit) and a new data sample S_{ad}, summarized in a correcting block NB (the value of the output parameter $t(s)$ is calculated by correcting block NB in the case, if the s instance is characterized by a high degree of belonging $\bigcup_{j=1}^{N_{RNB}} \mu_{NBj}^{(2)} > \mu_{\min}$ to the rules of the new structural element).

The error E_{Str} value on the sample S_{tr} for the BPS method (ranges from 0.0237 to 0.0299) is similar to the error value E_S. The error E_{Str} calculated on the basis of the sample S_{tr} using the model built using BPSad method is high enough for small values of the variable ω ($E_{Str} = 0.0945$ with $\omega = 1\%$, $E_{Str} = 0.0649$ with $\omega = 10\%$, $E_{Str} = 0.0592$ with $\omega = 20\%$). This is an unacceptable result, which is justified by the using of sample S_{ad} instances when building a new model NFNN. Error E_{Str} when using the method MATDNFM is quite low, including at small values of the index ω ($E_{Str} = 0.0299$ with $\omega = 1\%$, $E_{Str} = 0.0324$ with $\omega = 10\%$, $E_{Str} = 0.0294$ with $\omega = 100\%$). Such values E_{Str} confirm that the proposed method is appropriate to use at low values of ω, that is, in cases where the volume of new information S_{ad} about the objects or processes is significantly lower than the amount of available information S_{tr} that was used to build the basic model of NFN.

The table shows that the value of the error E_{Sad} calculated on the basis of the sample S_{ad} at small values of the variable ω (1% and 10%) is zero for all methods (except for the value $E_{Sad} = 0.0556$ for the BPS method with $\omega = 10\%$), what indicates their ability to implement new data into the existing model. However, comparing the values of the values E_{Sad}, E_{Str} and E_S, we can conclude that the method BPSad, in contrast to the proposed method MATDNFM, loses its ability to approximate the model (the value E_S increases to an unacceptable value at low levels of value of ω). The BPS method

DOI 10.15588/1607-3274-2018-3-12

is similar to the proposed method and provides synthesis (training) of neuro-fuzzy models with acceptable approximating properties ($E_S = 0.0296$), but the time before additional training (re-synthesis) of the model using the BPS method is high enough ($t_{ad} = 82.632$ sec.) and commensurate with the training time of the basic model that, unlike the proposed method MATDNFM ($t_{ad} = 0.6918$ sec. with $\omega = 1\%$ and $t_{ad} = 25.893$ sec. with $\omega = 100\%$, what is significantly less than when using the method BPS), significantly limits its use in practice, especially when processing big data.

Low values of the error E_t of model $NFNN$ on test data (with the exception of the method BPSad at low values of the variable ω), calculated for the test sample data (data about observations, which are not reflected in the sample $S = < P, T >$) confirm the ability to the generalization data by neuro-fuzzy diagnosis models which passed the additional training process .

Thus, the proposed method of MATDNFM is advisable to use at low values of ω , that is, in cases where the volume of new information S_{ad} about the objects or processes is significantly lower than the amount of available information S_{tr} that was used to build the basic model of NFN.

Given that the number of new data (variable ω) is usually significantly lower than the amount of initial information when solving practical problems, the use of the proposed method is appropriate, since the model $NFNN$ error E_S on the sample $S = < P, T >$ does not change almost when the values of the variable ω change, and the additional training time is much less than when using the BPS method.

CONCLUSIONS

The urgent problem of automation of the process of assessing the informativeness of features in solving problems of diagnosing and pattern recognition has been solved.

The scientific novelty of obtained results is that the method has been developed for additional training of diagnostic neuro-fuzzy models, which allows to adapt existing models to the change in the functioning environment by modifying them taking into account the information obtained as a result of new observations. The proposed method assumes the stages of extraction and grouping of correcting specimens, diagnosing with the help of the existing model leads to incorrect results, as well as the construction of a correcting block that summarizes the data of the correcting instances and its introduction into an already existing model. When determining the adjustable parameters of the correction block in the developed method, it is proposed to use information about the values of the coordinates of the correcting instances, as well as information on the degree of their belonging to clusters in the feature space (and, accordingly, to the fuzzy rules presented in the correcting block). This allows one to take into account the importance of corrective copies for restoring the functions of the fourth layer of the correcting block and, when determining custom parameters, to increase the contribution of those specimens that are characterized by high estimates of the degree of belonging to a particular cluster. Using the proposed method of learning the diagnostic neural-fuzzy models allows not to perform the resource-intensive process of re-constructing the diagnostic model on the basis of a complete set of data, to use the already existing model as the computing unit of the new model. In addition, models synthesized using the proposed method are highly interpretive, since each block generalizes information about its data set and uses neuro-fuzzy models as a basis.

The practical significance of obtained results is that the practical tasks of diagnosing and recognizing images are solved. The results of the experiments showed that the proposed method makes it possible to carry out additional training of diagnostic neuro-fuzzy models on the basis of new information and can be used in practice for solving practical problems of diagnosing and pattern recognition.

Prospects for further research are to develop the new methods for the additional training of deep learning neural networks for the big data processing

ACKNOWLEDGEMENTS

The work is supported by the state budget scientific research project of software tools department of Zaporizhzhya National Technical University “Methods and means of decision-making for data processing in intellectual recognition systems” (state registration number 0117U003920).

REFERENCES

1. Suzuki K. Artificial Neural Networks: Architectures and Applications. New York, InTech, 2013, 264 p. DOI: 10.5772/3409.
2. Hanrahan G. Artificial Neural Networks in Biological and Environmental Analysis. Boca Raton, Florida, CRC Press, 2011, 214 p. DOI: 10.1201/b10515.
3. Price C. Computer based diagnostic systems. London, Springer, 1999, 136 p. DOI: 10.1007/978-1-4471-0535-0.
4. Nauck D., Klawonn F., Kruse R. Foundations of neuro-fuzzy systems. Chichester, John Wiley & Sons, 1997, 305 p.
5. Oliinyk A., Skrupsky S., Subbotin S., Blagodariov O., Gofman Ye. Parallel computing system resources planning for neuro-fuzzy models synthesis and big data processing, *Radio Electronics, Computer Science, Control,* 2016, Vol. 4, pp. 61–69. DOI: 10.15588/1607-3274-2016-4-8.
6. Bow S. Pattern recognition and image preprocessing. New York, Marcel Dekker Inc., 2002, 698 p. DOI: 10.1201/9780203903896.
7. eds. C. Sammut, Webb G. I. Encyclopedia of machine learning. New York, Springer, 2011, 1031 p. DOI: 10.1007/978-0-387-30164-8.
8. ed. Ch. L. Mumford Computational intelligence: collaboration, fusion and emergence. New York, Springer, 2009, 752 p. DOI: 10.1007/978-3-642-01799-5.
9. Ding S. X. Model-based fault diagnosis techniques: design schemes, algorithms, and tools. Berlin, Springer, 2008, 473 p. DOI: 10.1007/978-1-4471-4799-2.

DOI 10.15588/1607-3274-2018-3-12

10. Shin Y. C., Xu C. Intelligent systems : modeling, optimization, and control. Boca Raton, CRC Press, 2009, 456 p. DOI: 10.1201/9781420051773.
11. Oliinyk A. Production rules extraction based on negative selection, *Radio Electronics, Computer Science, Control,* 2016, No. 1, pp. 40–49. DOI: 10.15588/1607-3274-2016-1-5.
12. Kolpakova T., Oliinyk A., Lovkin V. Integrated method of extraction, formalization and aggregation of competitive agents expert evaluations in group, *Radio Electronics, Computer Science, Control*, 2017, No. 2, pp. 100–108. DOI: 10.15588/1607-3274-2017-2-11.
13. eds. Berthold M., Hand D. J. Intelligent data analysis: an introduction. New York, Springer Verlag, 2007, 525 p. DOI: 10.1007/978-3-540-48625-1.
14. Tenne Y., Goh C.-K. Computational Intelligence in Expensive Optimization Problems. Berlin, Springer, 2010, 800 p. DOI: 10.1007/978-3-642-10701-6.
15. Oliinyk A. A. Skrupsky S. Yu. , Shkarupylo V. V. , Subbotin S. A. The model for estimation of computer system used resources while extracting production rules based on parallel computations, *Radio Electronics, Computer Science, Control,* 2017, No. 1, pp. 142–152. DOI: 10.15588/1607-3274-2017-1-16.
16. Oliinyk A. A., Skrupsky S. Yu. , Shkarupylo V. V. , Blagodariov O. Parallel multiagent method of big data reduction for pattern recognition, *Radio Electronics, Computer Science, Control,* 2017, No. 2, pp. 82–92. DOI: 10.15588/1607-3274-2017-2-9.
17. Bishop C. Neural Networks for pattern recognition. New York, Oxford University Press, 1995, 482 p.
18. Blanke M., Kinnaert M., Lunze J., Staroswiecki M. Diagnosis and fault-tolerant control. Berlin, Springer, 2006, 672 p. DOI: 10.1007/978-3-662-47943-8.
19. Oliinyk A. A., Subbotin S. A., Skrupsky S. Yu. , Lovkin V. M. , Zaiko T. A. Information Technology of Diagnosis Model Synthesis Based on Parallel Computing, *Radio Electronics, Computer Science, Control*, 2017, No. 3, pp. 139–151. DOI: 10.15588/1607-3274-2017-3-16.
20. Vachtsevanos G. , Lewis F., Roemer M. et al. Intelligent fault diagnosis and prognosis for engineering systems. New Jersey, John Wiley & Sons, 2006, 434 p. DOI: 10.1002/9780470117842.
21. Jang J. R., Sun C.-T., Mizutani E. Neuro-fuzzy and soft computing: a computational approach to learning and machine intelligence. Upple Saddle River: Prentice-Hall, 1997, 614 p. DOI: 10.1109/TAC.1997.633847.
22. Oliinyk A. Subbotin S., Lovkin V. , Ilyashenko M., Blagodariov O. Parallel method of big data reduction based on stochastic programming approach, *Radio Electronics, Computer Science, Control,* 2018, No. 2, pp. 60–72.
23. Shitikova O. V., Tabunshchyk G. V. // Method of Managing Uncertainty in Resource-Limited Settings, *Radio Electronics, Computer Science, Control,* 2015, No. 2, pp. 87–95. DOI: 10.15588/1607-3274-2015-2-11.
24. Rutkowski L. Flexible neuro-fuzzy systems : structures, learning and performance evaluation. Boston, Kluwer, 2004. 276 p. DOI: 10.1109/TNN.2003.811698.
25. Jensen R., Shen Q. Computational intelligence and feature selection: rough and fuzzy approaches. Hoboken, John Wiley & Sons, 2008, 339 p. DOI: 10.1002/9780470377888.
26. Oliinyk A., Subbotin S., Lovkin V., Blagodariov O., Zaiko T. The System of Criteria for Feature Informativeness Estimation in Pattern Recognition, *Radio Electronics, Computer Science, Control,* 2017, No. 4, pp. 85–96. DOI: 10.15588/1607-3274-2017-4-10.
27. Mulaik S. A. Foundations of Factor Analysis. Boca Raton, Florida, CRC Press, 2009, 548 p.
28. Sobhani-Tehrani E., Khorasani K. Fault diagnosis of nonlinear systems using a hybrid approach. New York, Springer, 2009, 265 p. (Lecture notes in control and information sciences ; № 383). DOI: 10.1007/978 -0-387-92907-1.
29. Salfner F., Lenk M., Malek M. A survey of online failure prediction methods, *ACM computing surveys*, 2010, Vol. 42, Issue 3, pp. 1–42. DOI: 10.1145/1670679.1670680.
30. Subbotin S., Oliinyk A., Skrupsky S. Individual prediction of the hypertensive patient condition based on computational intelligence / S. Subbotin, // I*nformation and Digital Technologies : International Conference IDT'2015, Zilina, 7–9 July 2015 : proceedings of the conference.* Zilina, Institute of Electrical and Electronics Engineers, 2015, pp. 336–344. DOI: 10.1109/DT.2015.7222996.

Received 25.06.2018.
Accepted 02.08.2018.

УДК 004.272.26: 004.93

ДОНАВЧАННЯ ДІАГНОСТИЧНИХ НЕЙРО-НЕЧІТКИХ МОДЕЛЕЙ

Олійник А. О. – канд. техн. наук, доцент, доцент кафедри програмних засобів Запорізького національного технічного університету, Запоріжжя, Україна.

Субботін С. О. – д-р. техн. наук, професор, завідувач кафедри програмних засобів Запорізького національного технічного університету, Запоріжжя, Україна.

Леощенко С. Д. – студент кафедри програмних засобів Запорізького національного технічного університету, Запоріжжя, Україна.

Ильяшенко М. Б. – канд. техн.наук, доцент, доцент кафедри комп'ютерних систем та мереж Запорізького національного технічного університету, Запоріжжя, Україна.

Миронова Н. О. – канд. техн. наук, доцент кафедри програмних засобів Запорізького національного технічного університету, Запоріжжя, Україна.

Мастиновський – канд. фіз.-мат. наук, професор, завідувач кафедри прикладної математики Запорізького національного технічного університету, Запоріжжя, Україна.

DOI 10.15588/1607-3274-2018-3-12

АНОТАЦІЯ

Актуальність. Вирішено задачу автоматизації синтезу діагностичних моделей при діагностуванні та розпізнаванні образів. Об’єкт дослідження – методи синтезу нейро-нечітких діагностичних моделей. Предмет дослідження – методи донавчання нейро-нечітких мереж.. Мета роботи – створення методу донавчання нейро-нечітких діагностичних моделей.

Метод. Запропоновано метод донавчання діагностичних нейро-нечітких моделей, який дозволяє адаптувати до зміни середовища функціонування існуючі моделі шляхом їх модифікації з урахуванням інформації, отриманої в результаті нових спостережень. Даний метод передбачає виконання етапів видобування та угрупування коригувальних екземплярів, діагностування за якими за допомогою існуючої моделі призводить до некоректних результатів, а також побудову коригувального блоку, який узагальнює дані коригувальних екземплярів, і впровадження його у вже існуючу модель. Використання запропонованого методу донавчання діагностичних нейро-нечітких моделей дозволяє не виконувати ресурсноміcткий процес повторної побудови діагностичної моделі на основі повного набору даних, використовувати вже наявну модель в якості обчислювального блоку нової моделі. Моделі, синтезовані за допомогою запропонованого методу, характеризуються високою інтерпретовністю, оскільки кожен блок узагальнює інформацію про свій набір даних і в якості базису використовує нейро-нечіткі моделі.

Результати. Розроблено програмне забезпечення, яке реалізує запропонований метод донавчання нейро-нечітких мереж і дозволяє виконувати перебудову існуючих діагностичних моделей на основі нової інформації про досліджувані об’єкти або процеси.

Висновки. Проведені експерименти підтвердили працездатність запропонованого метод донавчання нейро-нечітких мереж і дозволяють рекомендувати його для використання на практиці при обробці масивів даних для діагностування та розпізнавання образів. Перспективи подальших досліджень можуть полягати в розробці нових методів донавчання глибоких нейромереж для оброблення великих даних.

КЛЮЧОВІ СЛОВА: вибірка даних, діагностування, донавчання, нейро-нечітка модель, параметр, функція належності.

УДК 004.272.26: 004.93

ДООБУЧЕНИЕ ДИАГНОСТИЧЕСКИХ НЕЙРО-НЕЧЕТКИХ МОДЕЛЕЙ

Олейник А. А. – канд. техн. наук, доцент, доцент кафедры программных средств Запорожского национального технического университета, Запорожье, Украина.

Субботин С. А. – д-р техн. наук, профессор, заведующий кафедрой программных средств Запорожского национального технического университета, Запорожье, Украина.

Леощенко С. Д. – студент кафедры программных средств Запорожского национального технического университета, Запорожье, Украина.

Ильяшенко М. Б. – канд. техн. наук, доцент, доцент кафедры компьютерных систем и сетей Запорожского национального технического университета, Запорожье, Украина.

Миронова Н. А. – канд. техн. наук, доцент кафедры программных средств Запорожского национального технического университета, Запорожье, Украина.

Мастиновский Ю. В. – канд. физ.-мат. наук, профессор, заведующий кафедрой прикладной математики Запорожского национального технического университета, Запорожье, Украина.

АННОТАЦИЯ

Актуальность. Решена задача автоматизации синтеза диагностических моделей при диагностировании и распознавании образов. Объект исследования – методы синтеза нейро-нечетких диагностических моделей. Предмет исследования – методы дообучения нейро-нечетких сетей. Цель работы – создание метода дообучения нейро-нечетких диагностических моделей.

Метод. Предложен метод дообучения диагностических нейро-нечетких моделей, который позволяет адаптировать к изменению среды функционирования существующие модели путем их модификации с учетом информации, полученной в результате новых наблюдений. Данный метод предусматривает выполнение этапов извлечения и группировки корректирующих экземпляров, диагностирование по которым с помощью существующей модели приводит к некорректным результатам, а также построение корректирующего блока, который обобщает данные корректирующих экземпляров, и его внедрение в уже существующую модель. Использование предложенного метода дообучения диагностических нейро-нечетких моделей позволяет не выполнять ресурсоемкий процесс повторного построения диагностической модели на основе полного набора данных, использовать уже имеющуюся модель в качестве вычислительного блока новой модели. Модели, синтезированные с помощью предложенного метода, характеризуются высокой интерпретабельностью, поскольку каждый блок обобщает информацию о своем наборе данных и в качестве базиса использует нейро-нечеткие модели.

Результаты. Разработано программное обеспечение, реализующее предложенный метод дообучения нейро-нечетких сетей и позволяющее выполнять перенастройку существующих диагностических моделей на основе новой информации об исследуемых объектах или процессах.

Выводы. Проведенные эксперименты подтвердили работоспособность предложенного метода дообучения нейро-нечетких сетей и позволяют рекомендовать его для использования на практике при обработке массивов данных для диагностирования и распознавания образов. Перспективы дальнейших исследований могут заключаться в разработке новых методов дообучения глубоких нейросетей для обработки больших данных.

КЛЮЧЕВЫЕ СЛОВА: выборка данных, диагностирование, дообучение, нейро-нечеткая модель, параметр, функция принадлежности.

DOI 10.15588/1607-3274-2018-3-12

ЛІТЕРАТУРА / ЛИТЕРАТУРА

1. Suzuki K. Artificial Neural Networks: Architectures and Applications / K. Suzuki. – New York : InTech, 2013. – 264 p. DOI: 10.5772/3409.
2. Hanrahan G. Artificial Neural Networks in Biological and Environmental Analysis / G. Hanrahan. – Boca Raton, Florida : CRC Press, 2011. – 214 p. DOI: 10.1201/b10515.
3. Price C. Computer based diagnostic systems / C. Price. – London: Springer, 1999. – 136 p. DOI: 10.1007/978-1-4471-0535-0.
4. Nauck D. Foundations of neuro-fuzzy systems / D. Nauck, F. Klawonn, R. Kruse. – Chichester : John Wiley & Sons, 1997. – 305 p.
5. Oliinyk A. Parallel computing system resources planning for neuro-fuzzy models synthesis and big data processing / [A. Oliinyk, S. Skrupsky, S. Subbotin, O. Blagodariov] // Radio Electronics, Computer Science, Control. – 2016. – № 4. – P. 61–69. DOI: 10.15588/1607-3274-2016-4-8.
6. Bow S. Pattern recognition and image preprocessing / S. Bow. – New York : Marcel Dekker Inc., 2002. – 698 p. DOI: 10.1201/9780203903896.
7. Encyclopedia of machine learning / [eds. C. Sammut, G. I. Webb]. – New York : Springer, 2011. – 1031 p. DOI: 10.1007/978-0-387-30164-8.
8. Computational intelligence: collaboration, fusion and emergence / [ed. Ch. L. Mumford]. – New York : Springer, 2009. – 752 p. DOI: 10.1007/978-3-642-01799-5.
9. Ding S. X. Model-based fault diagnosis techniques: design schemes, algorithms, and tools / S. X. Ding. – Berlin : Springer, 2008. – 473 p. DOI: 10.1007/978-1-4471-4799-2.
10. Shin Y.C. Intelligent systems : modeling, optimization, and control / C. Y. Shin, C. Xu. – Boca Raton: CRC Press, 2009. – 456 p. DOI: 10.1201/9781420051773.
11. Oliinyk A. Production rules extraction based on negative selection / A. Oliinyk // Radio Electronics, Computer Science, Control. – 2016. – № 1. – P. 40–49. DOI: 10.15588/1607-3274-2016-1-5.
12. Kolpakova T. Integrated method of extraction, formalization and aggregation of competitive agents expert evaluations in group / T. Kolpakova A. Oliinyk, V. Lovkin // Radio Electronics, Computer Science, Control. – 2017. –№ 2. – P. 100–108. DOI: 10.15588/1607-3274-2017-2-11.
13. Intelligent data analysis: an introduction / [eds. M. Berthold, D. J. Hand]. – New York: Springer Verlag, 2007. – 525 p. DOI: 10.1007/978-3-540-48625-1.
14. Tenne Y. Computational Intelligence in Expensive Optimization Problems / Y. Tenne, C.-K. Goh. – Berlin : Springer: 2010. – 800 p. DOI: 10.1007/978-3-642-10701-6.
15. The model for estimation of computer system used resources while extracting production rules based on parallel computations / [A. A. Oliinyk, S. Yu. Skrupsky, V. V. Shkarupylo, S. A. Subbotin] // Radio Electronics, Computer Science, Control. – 2017. – № 1. – C. 142–152. DOI: 10.15588/1607-3274-2017-1-16.
16. Parallel multiagent method of big data reduction for pattern recognition / [A. A. Oliinyk, S. Yu. Skrupsky, V. V. Shkarupylo, O. Blagodariov] // Radio Electronics, Computer Science, Control. – 2017. – № 2. – C. 82–92. DOI: 10.15588/1607-3274-2017-2-9.
17. Bishop C. Neural Networks for pattern recognition / C. Bishop. – New York : Oxford University Press, 1995. – 482 p.
18. Diagnosis and fault-tolerant control / [M. Blanke, M. Kinnaert, J. Lunze, M. Staroswiecki]. – Berlin: Springer, 2006. – 672 p. DOI: 10.1007/978-3-662-47943-8.
19. Information Technology of Diagnosis Model Synthesis Based on Parallel Computing / [A. A. Oliinyk, S. A. Subbotin, S. Yu. Skrupsky et al.] // Radio Electronics, Computer Science, Control. – 2017. – № 3. – P. 139–151. DOI: 10.15588/1607-3274-2017-3-16.
20. Intelligent fault diagnosis and prognosis for engineering systems / [G. Vachtsevanos, F. Lewis, M. Roemer et al.]. – New Jersey: John Wiley & Sons, 2006. – 434 p. DOI: 10.1002/9780470117842.
21. Jang J. R. Neuro-fuzzy and soft computing: a computational approach to learning and machine intelligence / J. R. Jang, C.-T. Sun, E. Mizutani. – Upple Saddle River: Prentice-Hall, 1997. – 614 p. DOI: 10.1109/TAC.1997.633847.
22. Parallel method of big data reduction based on stochastic programming approach / [A. Oliinyk, S. Subbotin, V. Lovkin et al.] // Radio Electronics, Computer Science, Control. – 2018. – № 2. – P. 60–72.
23. Shitikova O. V. Method of Managing Uncertainty in Resource-Limited Settings / O. V. Shitikova, G. V. Tabunshchyk // Radio Electronics Computer Science Control. – 2015. – № 2. – P. 87–95. DOI: 10.15588/1607-3274-2015-2-11.
24. Rutkowski L. Flexible neuro-fuzzy systems : structures, learning and performance evaluation / L. Rutkowski. – Boston: Kluwer, 2004. – 276 p. DOI: 10.1109/TNN.2003.811698.
25. Jensen R. Computational intelligence and feature selection: rough and fuzzy approaches / R. Jensen, Q. Shen. – Hoboken: John Wiley & Sons, 2008. – 339 p. DOI: 10.1002/9780470377888.
26. Oliinyk A. A The System of Criteria for Feature Informativeness Estimation in Pattern Recognition / [A. Oliinyk, S. Subbotin, V. Lovkin et al] // Radio Electronics, Computer Science, Control. – 2017. – № 4. – P. 85–96. DOI: 10.15588/1607-3274-2017-4-10.
27. Mulaik S. A. Foundations of Factor Analysis / S. A. Mulaik. – Boca Raton, Florida: CRC Press. – 2009. – 548 p.
28. Sobhani-Tehrani E. Fault diagnosis of nonlinear systems using a hybrid approach / E. Sobhani-Tehrani, K. Khorasani. – New York: Springer, 2009. – 265 p. – (Lecture notes in control and information sciences ; № 383). DOI: 10.1007/978-0-387-92907-1.
29. Salfner F. A survey of online failure prediction methods / F. Salfner, M. Lenk, M. Malek // ACM computing surveys. – 2010. – Vol. 42, Issue 3. – P. 1–42. DOI: 10.1145/1670679.1670680.
30. Subbotin S. Individual prediction of the hypertensive patient condition based on computational intelligence / S. Subbotin, A. Oliinyk, S. Skrupsky // Information and Digital Technologies : International Conference IDT'2015, Zilina, 7–9 July 2015 : proceedings of the conference. – Zilina : Institute of Electrical and Electronics Engineers, 2015. – P. 336–344. DOI: 10.1109/DT.2015.7222996.

DOI 10.15588/1607-3274-2018-3-12