Neural Network Modeling Method of Transformations Data of Audit Production with Returnable Waste

Tatiana Neskorodieva[1], Eugene Fedorov[1,2], Oleksii Smirnov[3] and Pavlo Rymar[1]

[1] *Vasyl' Stus Donetsk National University, 600-richchia str., 21, Vinnytsia, 21021, Ukraine*
[2] *Cherkasy State Technological University, Shevchenko blvd., 460, Cherkasy, 18006, Ukraine*
[3] *Central Ukrainian National Technical University, avenue University, 8, Kropivnitskiy, 25006, Ukraine*

Abstract

Currently, the analytical procedures used during the audit are based on data mining techniques. The object of the research is the process of the content auditing of the production with returnable waste and intermediate products. The aim of the work is to reduce the risk of incorrect display of the dataset in the DSS of the audit of the method of neural network modeling of transformations of audit data of production with recyclable waste and intermediates. This will reduce the risk of the validated data misclassification. Audit data set transformations of a prerequisite "Completeness" are presented the sequences of sets data mappings of consecutive operations. Reached further development a method of parametrical identification of the MRMLP model which considers number of iterations of training and combines Gaussian distributions and Cauchy that increases the forecast accuracy as on initial iterations all search space is investigated, and on final iterations the search becomes directed. The software implementing the offered methods in MATLAB package was developed and investigated on the data of the release of raw materials into production and the posting of finished products of a with a two-year depth of sampling with daily time intervals. The made experiments confirmed operability of the developed software and allow to recommend it for use in practice in a subsystem of the automated analysis of DSS of audit for check of maps of sets of data of the raw materials release into production and the products output.

Keywords

production audit, returnable waste, intermediate products, mapping by neural network, modified recurrent multilayered perceptron, metaheuristics, DSS, risk of wrong mapping of data sets, risk of the validated data misclassification.

1. Introduction

In the process of development of international and national economics, industry of IT, it is possible to distinguish the following basic tendencies: digital transformations realization, digital economy forming, socio-economic processes globalization and IT accompanying them [1].

CITRisk'2021: 2nd International Workshop on Computational & Information Technologies for Risk-Informed Systems, September 16–17, 2021, Kherson, Ukraine
EMAIL: t.neskorodieva@donnu.edu.ua (T.Neskorodieva); fedorovee75@ukr.net (E.Fedorov); Dr.smirnovoa@gmail.com (O.Smirnov); p.rymar@donnu.edu.ua (P.Rymar)
ORCID: 0000-0003-2474-7697 (T.Neskorodieva); 0000-0003-3841-7373 (E.Fedorov); 0000-0001-9543-874X (O.Smirnov); 0000-0002-0647-2020 (P.Rymar)

These processes result in the origin of global, multilevel hierarchical structures of heterogeneous, multivariable, multifunction connections, interactions and cooperation of manage objects (objects of audit). Large volumes of information about them have been accumulated in the information systems of account, management and audit.

Consequently, nowadays the scientific and technical issue of the modern information technologies in financial and economic sphere of Ukraine is forming of the methodology of planning and creation of the decision support systems (DSS) at the audit of enterprises in the conditions of application of IT and with the use of information technologies on the basis of the automated analysis of the large volumes of data about financial and economic activity and states of enterprises with the multi-level hierarchical structure of heterogeneous, multivariable, multifunction connections, intercommunications and cooperation of objects of audit with the purpose of expansion of functional possibilities, increase of efficiency and universality of IT audit [2, 3].

Currently, analytical procedures used during the audit are based on data mining techniques [4-6]. Automated DSS audit means the automatic forming of recommendable decisions, based on the results of the data automated analysis, that improves quality process of audit and reducing the risk of incorrect display of datasets [7, 8]. Unlike the traditional approach, computer technologies of analysis of data in the audit system accelerate and promote the process accuracy of audit, that extremely critical in the conditions of plenty of associate tasks on lower and middle levels and amounts of indexes and supervisions in every task [9].

When developing a decision-making system in audit based on data mining technologies, three methods have been created: classifying variables, forming analysis sets, mapping analysis sets.

The peculiarity of the methodology for classifying indicators is that qualitatively different (by semantic content) variables are classified: numerological, linguistic, quantitative, logical. The essence of the second technique is determined by the qualitative meaning of the indicators. In accordance with this, sets are formed with the corresponding semantic content: document numbers, name of indicators, quantitative estimates of the values of indicators, logical indicators.

The third technique is subordinated to the mappings of formed sets of the same type on each other to determine equivalence in the following senses: numerological, linguistic, quantitative, logical.

For modeling of data transformations of audit of production neural networks.

The following are most often used as neural networks for mapping audit indicators:

– Elman's (ENN) neural network or a simple recurrent network (SRN) [10, 11] which is a recurrent two-layer network and is constructed based on MLP. The advantage of this network is simpler architecture and higher speed of training, than in gated, reservoir and bidirectional networks. A disadvantage is the insufficient accuracy of the forecast in comparison with gated, reservoir and bidirectional networks;

– the bidirectional recurrent neural network (BRNN) [12, 13] which is a recurrent two-layer network and is constructed based on two neural networks of Elman. The advantage of this network is higher forecast accuracy, than in a normal neural network of Elman. A disadvantage is higher complexity of determination of architecture, lower speed of training, than in a normal neural network of Elman;

– long short-term memory (LSTM) [14, 15] which is a recurrent network and is constructed based on memory units (contain one or more cells), and input, output, a forget of gates (FIRs filters). The advantage of this network is higher forecast accuracy, than in a normal neural network of Elman. A disadvantage is higher complexity of architecture determination, lower training speed, than in a normal neural network of Elman;

– the bidirectional long short-term memory (BLSTM) [16, 17] which is a recurrent network and is constructed based on two neural networks of LSTM. The advantage of this network is higher forecast accuracy, than in normal LSTM. A disadvantage is higher complexity of architecture determination, lower speed of training, than in normal LSTM;

– the gated recurrent unit (GRU) [18, 19] which is a recurrent two-layer network and is constructed based on the hidden unit's gates of reset and update (FIRs filters). The advantage of this network is higher accuracy of the forecast, than in a normal neural network of Elman. A disadvantage is higher complexity of architecture determination, lower training speed, than in a normal neural network of Elman;

– the echo state network (ESN) [20] which is a recurrent two-layer network is constructed based on the reservoir (represents a layer of the interconnected not full-connected neurons). The advantage of this network is higher forecast accuracy, than in a normal neural network of Elman. A disadvantage is higher complexity of architecture determination, lower training speed, than in a normal neural network of Elman;

– the liquid state machine (LSM) [21] which is a recurrent two-layer network is constructed based on the reservoir (represents a layer of the interconnected not full-connected spike neurons) and MLP. The advantage of this network is higher forecast accuracy, than in a normal neural network of Elman. A disadvantage is higher complexity of architecture determination, lower training speed, than in a normal neural network of Elman.

Thus, any of networks does not meet all criteria.

For acceleration of training and increase an accuracy of data transformations model of production audit now are used metaheuristics (or modern heuristics) [22]. The metaheuristics expands opportunities heuristic, combining heuristic methods based on high-level strategy [23].

Existing metaheuristics possess one or more of the following disadvantages:

– there is only description abstraction of a method or the method description is focused on the solution only of a certain task [24];

– influence of iteration number on solution search process the is not considered [25];

– the convergence of a method is not guaranteed [26];

– there is no opportunity to use not binary potential solutions [27];

– the procedure of parameters values determination is not automated [28];

– there is no opportunity to solve problems of conditional optimization [29];

– insufficient accuracy of a method [30].

In this regard there is a creation problem of effective metaheuristic methods of optimization.

In this regard, it is the actual to create a neural network that considers the functional structure of production with returnable and non-returnable waste and intermediate products and learns based on effective metaheuristics.

The aim of the work is to reduce the risk of incorrect display of the dataset in the audit DSS by the method of neural network modeling of audit data transformations of production with recyclable waste and intermediates.

For the objective achievement it is necessary to solve the following tasks:

– offer structural model of audit data transformations of production;

– offer neural network model of audit data transformations of production based on a recurrent multilayered perceptron;

– select criterion for evaluation of neural network model efficiency of production audit data transformation;

– offer a method of parametrical identification of neural network model of production audit data transformation based on the return distribution in time;

– offer a method of parametrical identification of neural network model of audit data transformation of production based on cross entropy and stochastic search of an extremum with training at vectors of normal distribution;

– execute numerical research.

The problem formulation. Let for model of data transformations of production audit the training set be set $S = \{(\mathrm{x}_\mu, \tilde{\mathrm{d}}_\mu, \widehat{\mathrm{d}}_\mu^{(1)}, ..., \widehat{\mathrm{d}}_\mu^{(H)})\}$, $\mu \in \overline{1,P}$, where x_μ is the μ-th training input vector, $\tilde{\mathrm{d}}_\mu$ is the μ -th training output vector of finished goods, $\widehat{\mathrm{d}}_\mu^{(k)}$ is the μ-th training output vector of unreturnable waste which are received after each k -th of a layer of production of semi-products.

Then a problem of increase an accuracy of production audit data transformations on model of the modified recurrent multilayered perceptron (MRLMP) $g(\mathrm{x},\mathrm{w})$, where x is an input signal, w is the parameters vector, is represented as the problem of finding such a model parameter vector w^* that satisfies the criterion

$$F = \frac{1}{P}\sum_{\mu=1}^{P}(g(\mathrm{x}_\mu, \mathrm{w}^*) - (\tilde{\mathrm{d}}_\mu, \widehat{\mathrm{d}}_\mu^{(1)}, ..., \widehat{\mathrm{d}}_\mu^{(H)}))^2 \to \min . \quad (1)$$

2. Materials and methods

2.1. Formalization of audit subject area data subelements transformations of the prerequisite "Completeness"

Audit data set transformations of a prerequisite "Completeness" will be presented the sequences of sets data mappings of consecutive operations

$$Z_{i_1} \to Z_{i_2} \to Z_{i_m} \to \ldots \to Z_{i_M},\ i_1 \prec \ldots \prec i_m \prec \ldots \prec i_M,\ (i_1,\ldots i_m,\ldots i_M) \in A(\tilde{\mathrm{I}}),\ \tilde{\mathrm{I}} = \overline{1,I}, \quad (2)$$

where Z is reporting data set,

$(i_1,\ldots i_m,\ldots i_M)$ is combination of consecutive operation types of a set $\tilde{\mathrm{I}} = \overline{1,I}$,

$A(\tilde{\mathrm{I}})$ is set of possible combinations on a set $\tilde{\mathrm{I}} = \overline{1,I}$.

Therefore "Completeness" prerequisite audit we will present as the transformations checking of subelements of data domain in the form of the sequences of mapping of splitting's data elements of the sequences

$$\Re(Z_{i_1}) \to \Re(Z_{i_2}) \to \Re(Z_{i_m}) \to \ldots \to \Re(Z_{i_M}),\ i_1 \prec \ldots \prec i_m \prec \ldots \prec i_M,\ (i_1,\ldots i_m,\ldots i_M) \in A(\tilde{\mathrm{I}}),\ \tilde{\mathrm{I}} = \overline{1,I}, \quad (3)$$

where $\Re(Z)$ is splitting set Z.

Possible combinations set of consecutive operation types $(i_1,\ldots i_m,\ldots i_M)$ defined in (3) includes check in direct and in the opposite direction.

The model of the subelements transformation of the "Completeness" prerequisite audit subject domain will be formed on the example of the direct material costs audit. Models of their conversions can be presented in the form of graphs in which everyone corresponds to a subelement, and an edge – to map which describes interrelation between the corresponding subelements.

For this purpose, we use formalization of a set of direct material costs in the form of the graph $G^{(1)} = (Z^{(1)}, R^{(1)})$ (fig. 1) where vertex – accounts on which account of these current assets is kept and edges are operations as a result of which there is their conversion. Then model of subelements conversion of audit data domain of a prerequisite "Completeness" at direct full check $((i_1, \dots i_m, \dots i_M) = (1,2,3,4))$ represents maps of subsets of these raw materials receipt $Z_{\Re}^{(r_1)}(i_1) \in \Re(Z_{i_1})$ release of raw materials in production $Z_{\Re}^{(r_2)}(i_2) \in \Re(Z_{i_2})$, then in subsets of production data $Z_{\Re}^{(r_3)}(i_3) \in \Re(Z_{i_3})$ receipt of finished goods $Z_{\Re}^{(r_4)}(i_4) \in \Re(Z_{i_4})$, $T \in \left\{ t_{j_m}, T_m,\ j = \overline{1, J_m}, m = \overline{1, M}, T \right\}$.

In that specific case, if splitting sets it was carried out on the basis of the logical conditions characterizing belonging to one of accounting item subspecies, then the model of subelements conversion of audit data domain of a prerequisite "Completeness" at direct full check is the set of the sequences of sets maps of these calculations operations for suppliers types in subsets of these operations on raw materials types, then in subsets of these operations on products types and finished goods types.

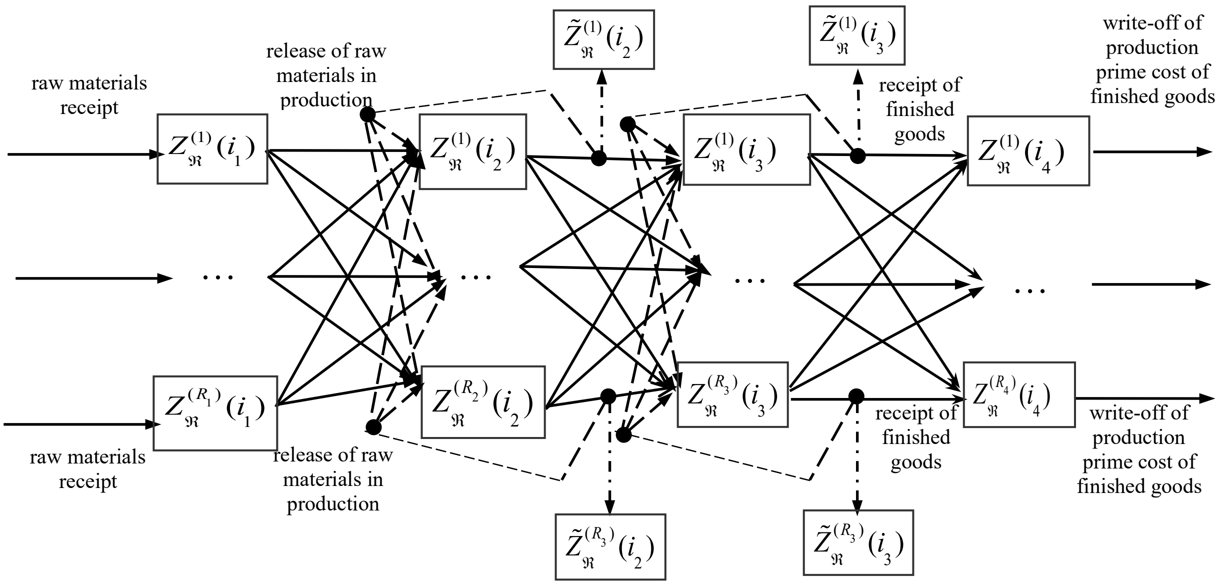

Figure 1: Model of conversions of audit data domain subelements of a prerequisite "Completeness"

2.2. Choosing a neural network model for mapping audit sets

The unit diagram of model of the modified recurrent multilayered perceptron (MRMLP) full-connected recurrent layers of semi-products production (the neurons forming them are designated in continuous white color), not full-connected non-recurrent layers of unreturnable waste (forming them neurons is presented on fig. 2 are designated in continuous black color) and not full-connected non-recurrent layer of finished goods (the neurons forming it are designated in continuous gray color).

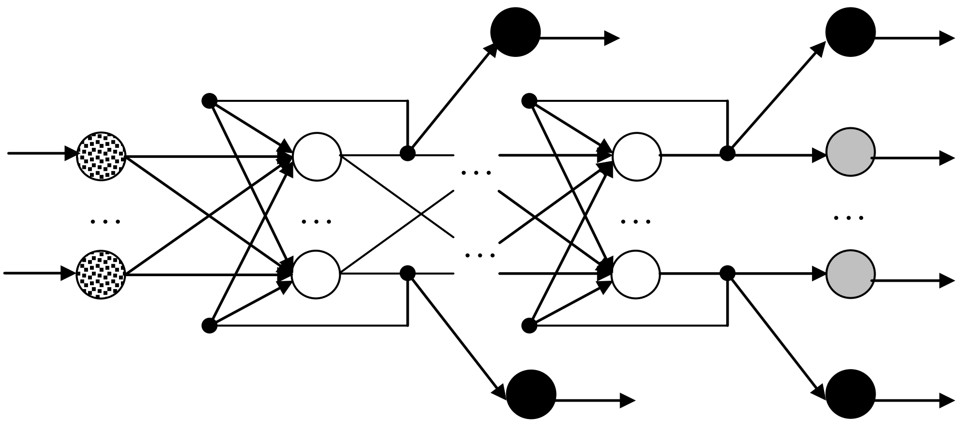

Figure 2: The unit diagram of the modified recurrent multilayered perceptron model

The MRMLP model, the executing map of each input sample of raw materials $\mathrm{x} = (x_1,...,x_{N^{(0)}})$, output samples of finished goods $\tilde{\mathrm{y}} = (\tilde{y}_1,...,\tilde{y}_{N^{(H)}})$ and unreturnable waste $\hat{\mathrm{y}}^{(1)} = (\hat{y}_1^{(1)},...,\hat{y}_{N^{(1)}}^{(1)})$, $\ldots$, $\hat{\mathrm{y}}^{(H)} = (\hat{y}_1^{(H)},...,\hat{y}_{N^{(H)}}^{(H)})$, it is presented in the form

$$y_i^{(0)}(n) = x_i,\ i \in \overline{1,N^{(0)}},$$

$$y_j^{(k)}(n) = f^{(k)}(s_j^{(k)}(n)),\ j \in \overline{1,N^{(k)}},\ k \in \overline{1,H},$$

$$s_j^{(k)}(n) = b_j^{(k)} + \sum_{i=1}^{N^{(k-1)}} w_{ij}^{(k)} y_i^{(k-1)}(n) + \sum_{i=1}^{N^{(k)}} \breve{w}_{ij}^{(k)} y_i^{(k)}(n-1),$$

$$\tilde{y}_j(n) = \tilde{f}(\tilde{s}_j(n)),\ j \in \overline{1,N^{(H)}},$$

$$\tilde{s}_j(n) = \tilde{b}_0(n) + \tilde{w}_{jj} y_i^{(H)}(n),$$

$$\hat{y}_j^{(k)}(n) = \hat{f}^{(k)}(\hat{s}_j^{(k)}(n)),\ j \in \overline{1,N^{(k)}},\ k \in \overline{1,H},$$

$$\hat{s}_j^{(k)}(n) = \hat{b}_0^{(k)}(n) + \hat{w}_{jj}^{(k)} y_j^{(k)}(n),$$

where $N^{(k)}$ – neurons number in k-th layer of semi-products production and unreturnable waste,

H – quantity of layers of semi-products production and unreturnable waste,

$N^{(0)}$ – number of neurons of an input layer (raw materials layer),

$b_j^{(k)}$ – bias for j- th of a neuron k-th of a layer of semi-products production,

$\hat{b}_j$ – bias for j- th of a neuron of a finished goods layer,

$\widehat{b}_j^{(k)}$ – bias for j -th of a neuron k -th of a the unreturnable remains layer,

$w_{ij}^{(k)}$ – communication weight from *i*-th of a neuron of k -th *1*-th layer of semi-products production to *j*-th to a neuron of k -th of a layer of semi-products production,

$\breve{w}_{ij}^{(k)}$ – communication weight from *i*-th of a neuron k -th of the semi-products production layer to j - th neuron of k -th *1*-th layer of semi-products production,

$\tilde{w}_{jj}$ – communication weight from j -th neuron *H*-th of a layer of semi-products production to j -th neuron of a finished goods layer,

$\widehat{w}_{jj}^{(k)}$ – communication weight from j -th of a neuron *k*-th of a layer of semi-products production to *j*-th neuron of k -th layer of irretrievable waste,

$y_j^{(k)}(n)$ – output of j -th neuron of k -th of a layer of semi-products production in timepoint n,

$\tilde{y}_j(n)$ – output of j -th finished goods layer in timepoint n,

$\widehat{y}_j^{(k)}(n)$ – output of j -th neuron of k -th of a irretrievable waste layer in timepoint n,

$f^{(k)}$ – neurons function neuron activation of k -th of a layer of semi-products production,

$\tilde{f}$ – neurons function activation of a finished goods layer,

$\widehat{f}^{(k)}$ – neurons function activation of k -th of a unreturnable waste layer.

2.3. Criterion choice for evaluation of neural network model efficiency of data transformations of production audit

In this work for training of the MRMLP model the function of the purpose which means the choice of such values of a vector of parameters is selected $\mathrm{w} = (w_{11}^{(1)}, ..., w_{N^{(H-1)}N^{(H)}}^{(H)}, \breve{w}_{11}^{(1)}, ..., \breve{w}_{N^{(H)}N^{(H-1)}}^{(H)}, \widehat{w}_{11}^{(1)}, ..., \widehat{w}_{N^{(H)}N^{(H)}}^{(H)}, \tilde{w}_{11}, ..., \tilde{w}_{N^{(H)}N^{(H)}})$, which deliver a minimum of a root mean square error (the differences of a sample on model and a test sample)

$$F = \frac{1}{PN^{(H)}} \sum_{\mu=1}^{P} \left\| \tilde{\mathrm{y}}_\mu - \tilde{\mathrm{d}}_\mu \right\|^2 + \frac{1}{HP} \sum_{k=1}^{H} \frac{1}{N^{(k)}} \sum_{\mu=1}^{P} \left\| \widehat{\mathrm{y}}_\mu^{(k)} - \widehat{\mathrm{d}}_\mu^{(k)} \right\|^2 \to \min_{\mathrm{w}},$$

where $\tilde{\mathrm{y}}_\mu, \mathrm{y}_\mu^{(1)}, ..., \mathrm{y}_\mu^{(H)}$ – μ -th output samples on model,

$\tilde{\mathrm{d}}_\mu, \widehat{\mathrm{d}}_\mu^{(1)}, ..., \widehat{\mathrm{d}}_\mu^{(H)}$ – μ -th test output samples,

H – quantity hidden layers,
P – power of a test set.

2.4. Method of parametrical identification of data transformations model of production audit based on the back propagation in time in a sequential mode

1. Number of iterations of training $n=1$, initialization by means of uniform distribution on an interval (0.1) or [-0.5, 0.5] bias $b_j^{(k)}(n)$, $\widehat{b}_j^{(k)}(n)$, $j \in \overline{1, N^{(k)}}, k \in \overline{1, H}$, $\widetilde{b}_j(n)$, $j \in \overline{1, N^{(H)}}$, and weights $w_{ij}^{(k)}(n)$, $i \in \overline{1, N^{(k-1)}}, j \in \overline{1, N^{(k)}}, k \in \overline{1, H}$, $\breve{w}_{ij}^{(k)}(n)$, $i \in \overline{1, N^{(k)}}, j \in \overline{1, N^{(k-1)}}, k \in \overline{1, H}$, $\widetilde{w}_{jj}(n)$, $j \in \overline{1, N^{(H)}}$, $\widehat{w}_{jj}^{(k)}(n)$, $j \in \overline{1, N^{(k)}}, k \in \overline{1, H}$.

2. The training set is set $\{(\mathrm{x}_\mu, \tilde{\mathrm{d}}_\mu, \widehat{\mathrm{d}}_\mu^{(1)}, ..., \widehat{\mathrm{d}}_\mu^{(H)}) \mid \mathrm{x}_\mu \in R^{N^{(0)}}, \tilde{\mathrm{d}}_\mu \in R^{N^{(H)}}, \widehat{\mathrm{d}}_\mu^{(k)} \in R^{N^{(k)}}\}$, $\mu \in \overline{1, P}$, where x_μ – μ-th the training input vector raw materials, $\tilde{\mathrm{d}}_\mu$ – μ-th the training output vector of finished goods, $\widehat{\mathrm{d}}_\mu^{(k)}$ – μ-th the training output vector of unreturnable waste which are received after each k-th of semi-products production layer, P – power of a training set. Number of the current train from a training set $\mu = 1$.

3. Initial calculation of a signal output for each full-connected recurrent hidden layer

$$y_i^{(k)}(n-1) = 0,\ i \in \overline{1, N^{(k)}},\ k \in \overline{1, H}.$$

4. Calculation of a signal output for each full-connected recurrent layer of semi-products production considering returnable waste (forward propagation)

$$y_i^{(0)}(n) = x_{\mu i},$$

$$y_j^{(k)}(n) = f^{(k)}(s_j^{(k)}(n)),\ j \in \overline{1, N^{(k)}},\ k \in \overline{1, H},$$

$$s_j^{(k)}(n) = b_j^{(k)}(n) + \sum_{i=1}^{N^{(k-1)}} w_{ij}^{(k)}(n) y_i^{(k-1)}(n) + \sum_{i=1}^{N^{(k)}} \breve{w}_{ij}^{(k)}(n) y_i^{(k)}(n-1).$$

5. Calculation of a signal output for not full-connected non-recurrent layer of finished goods (forward propagation)

$$\tilde{y}_j(n) = \tilde{f}(\tilde{s}_j(n)),\ j \in \overline{1, N^{(H)}},$$

$$\tilde{s}_j(n) = \tilde{b}_j(n) + \tilde{w}_{jj}(n) y_i^{(H)}(n).$$

6. Calculation of a signal output for each not full-connected non-recurrent layer of unretainable waste (forward propagation)

$$\widehat{y}_j^{(k)}(n) = \widehat{f}^{(k)}(\widehat{s}_j^{(k)}(n)),\ j \in \overline{1, N^{(k)}},\ k \in \overline{1, H},$$

$$\widehat{s}_j^{(k)}(n) = \widehat{b}_j^{(k)}(n) + \widehat{w}_{jj}^{(k)}(n) y_j^{(k)}(n).$$

7. Calculation of energy of error ANN

$$E(n) = \frac{1}{2} \sum_{j=1}^{N^{(H)}} \left(\tilde{e}_j(n)\right)^2 + \frac{1}{2} \sum_{k=1}^{H} \sum_{j=1}^{N^{(k)}} \left(\widehat{e}_j^{(k)}(n)\right)^2,$$

$\tilde{e}_j(n) = \tilde{y}_j(n) - \tilde{d}_{\mu j}$, $\widehat{e}_j^{(k)}(n) = \widehat{y}_j^{(k)}(n) - \widehat{d}_{\mu j}^{(k)}$.

8. Setup of synoptic weights based on generalized the delta rule (backward propagation)

$$b_j^{(k)}(n+1) = b_j^{(k)}(n) - \eta \frac{\partial E(n)}{\partial b_j^{(k)}(n)},\ j \in \overline{1, N^{(k)}},\ k \in \overline{1, H},$$

$$w_{ij}^{(k)}(n+1) = w_{ij}^{(k)}(n) - \eta \frac{\partial E(n)}{\partial w_{ij}^{(k)}(n)},\ i \in \overline{1, N^{(k-1)}},\ j \in \overline{1, N^{(k)}},\ k \in \overline{1, H},$$

$$\breve{w}_{ij}^{(k)}(n+1) = \breve{w}_{ij}^{(k)}(n) - \eta \frac{\partial E(n)}{\partial \breve{w}_{ij}^{(k)}(n)},\ j \in \overline{1, N^{(k)}},\ i \in \overline{1, N^{(k-1)}},\ k \in \overline{1, H},$$

$$\tilde{b}_j(n+1) = \tilde{b}_j(n) - \eta \frac{\partial E(n)}{\partial \tilde{b}_j(n)},\ j \in \overline{1, N^{(H)}},$$

$$\tilde{w}_{jj}(n+1) = \tilde{w}_{jj}(n) - \eta \frac{\partial E(n)}{\partial \tilde{w}_{jj}(n)},\ j \in \overline{1, N^{(H)}},$$

$$\widehat{b}_j^{(k)}(n+1) = \widehat{b}_j^{(k)}(n) - \eta \frac{\partial E(n)}{\partial \widehat{b}_j^{(k)}(n)},\ j \in \overline{1, N^{(k)}},\ k \in \overline{1, H},$$

$$\widehat{w}_{jj}^{(k)}(n+1) = \widehat{w}_{jj}^{(k)}(n) - \eta \frac{\partial E(n)}{\partial \widehat{w}_{jj}^{(k)}(n)},\ j \in \overline{1, N^{(k)}},\ k \in \overline{1, H},$$

where η is the parameter determining training speed (at big η training happens quicker, but the danger to receive the incorrect solution increases), $0 < \eta < 1$.

$$\frac{\partial E(n)}{\partial b_j^{(k)}(n)} = g_j^{(k)}(n),$$

$$\frac{\partial E(n)}{\partial w_{ij}^{(k)}(n)} = y_i^{(k-1)}(n) g_j^{(k)}(n),$$

$$\frac{\partial E(n)}{\partial \breve{w}_{ij}^{(k)}(n)} = y_i^{(k)}(n-1) g_j^{(k)}(n),$$

$$\frac{\partial E(n)}{\partial \tilde{b}_j(n)} = \tilde{g}_j(n),$$

$$\frac{\partial E(n)}{\partial \tilde{w}_{jj}(n)} = y_j^{(H)}(n) \tilde{g}_j(n),$$

$$\frac{\partial E(n)}{\partial \widehat{b}_j^{(k)}(n)} = \widehat{g}_j^{(k)}(n),$$

$$\frac{\partial E(n)}{\partial \widehat{w}_{jj}^{(k)}(n)} = y_j^{(k)}(n) \widehat{g}_j^{(k)}(n),$$

$$g_j^{(k)}(n)=\begin{cases} f'^{(H)}(s_j^{(H)}(n))\left(\tilde{w}_{jj}(n)\tilde{g}_j(n)+\hat{w}_{jj}^{(H)}(n)\hat{g}_j^{(H)}(n)\right), & k=H \\ f'^{(k)}(s_j^{(k)}(n))\left(\sum\limits_{l=1}^{N^{(k+1)}} w_{jl}^{(k+1)}(n)g_l^{(k+1)}(n)+\hat{w}_{jj}^{(k)}(n)\hat{g}_j^{(k)}(n)\right), & k<H \end{cases},$$

$\tilde{g}_j(n)=\tilde{f}'(\tilde{s}_j(n))\tilde{e}_j(n)$,

$\hat{g}_l^{(k)}(n)=\hat{f}'^{(k)}(\hat{s}_j^{(k)}(n))\hat{e}_j^{(k)}(n)$.

9. Check of a termination condition

If $n \bmod P>0$ then $\mu=\mu+1$, $n=n+1$, go to 4.

If $n \bmod P=0$ and $\frac{1}{P}\sum\limits_{s=1}^{P}E(n-P+s)>\varepsilon$ then $n=n+1$, go to 2.

If $n \bmod P=0$ and $\frac{1}{P}\sum\limits_{s=1}^{P}E(n-P+s)<\varepsilon$ to be completed.

A high probability of hit in a local extremum belongs to disadvantage of this method that reduces training accuracy, and impossibility of training in batch mode that reduces training speed. In this regard in work the alternative method of training at a basis metaheuristic is offered.

2.5. Method of parametrical identification of data transformations model of production audit on a basis metaheuristic

The offered method of parametrical identification of production audit data transformations model is based on a method of cross entropy and stochastic search of an extremum with training at vectors of normal distribution [30].

Feature of the offered method will be that for speed control of convergence of a method, speed control of change of distribution parameters and for providing that on initial iterations all search space was investigated, and on final iterations the search became directed, at generation of potential solutions number of iterations is considered. Besides, not only Gaussian distribution, but also Cauchy's distribution is used, and their value depends on number of iterations.

The offered method consists of the following stages:

1. Initialization

1.1. Task of the maximum number of iterations N, population size K, solution lengths M (corresponds to length of a vector of the model parameters MRMLP), the maximum quantity of the selected best solutions B, parameter for generation of scales parameters vector β, $0<\beta<1$.

1.2. Initialization of a location's parameters vector

$\gamma^{loc}=(\gamma_1^{loc},...,\gamma_M^{loc})$, $\gamma_j^{loc}=x_j^{\min}+\frac{1}{2}(x_j^{\max}-x_j^{\min})$.

1.3. Initialization of a location's parameters vector

$\gamma^{scale}=(\gamma_1^{scale},...,\gamma_M^{scale})$, $\gamma_j^{scale}=\beta(x_j^{\max}-x_j^{\min})$.

1.4. Define the best solution (best vector of the model parameters MRMLP)

$x^* = \gamma^{loc}$.

2. Iteration number $n = 1$.

3. Creation of the current population of potential solutions P.

3.1. Solution number $k = 1$, $P = \varnothing$.

3.2. Generation of the new potential solution x_k (vector of the model parameters MRMLP)

$$x_{kj} = \gamma_j^{loc} + \gamma_j^{scale}\left(\left(\frac{N-n}{N}\right)C(0,1) + \left(\frac{n}{N}\right)N(0,1)\right),\ j \in \overline{1,M},$$

where $N(0,1)$ is standard normal distribution,
$C(0,1)$ is standard Cauchy distribution.

3.3. If $k < K$ then $P = P \cup \{x_k\}$, $k = k + 1$, transition to a step 3.2.

4. Sort P on function of the purpose, i.e. $F(x_k) < F(x_{k+1})$.

5. Define the best solution (best vector of the model parameters MRMLP) on the current population

$$k^* = \arg\min_k F(x_k),\ k \in \overline{1,K}.$$

6. Define the best solution (best vector of the model parameters MRMLP) on all iterations

if $F(x_{k^*}) < F(x^*)$ then $x^* = x_{k^*}$.

7. Modification of distribution parameters (on a basis B the first, i.e. best, new potential solutions from population P).

7.1. Modification of a vector of parameters of locations

$$\gamma_j^{loc} = \left(\frac{n}{N}\right)\gamma_j^{loc} + \left(\frac{N-n}{N}\right)\tilde{\gamma}_j^{loc},\ \tilde{\gamma}_j^{loc} = \frac{1}{B}\sum_{k=1}^{B} x_{kj},\ j \in \overline{1,M}.$$

7.2. Modification of a vector of parameters of scales

$$\gamma_j^{scale} = \left(\frac{N-n}{N}\right)\beta(x_j^{\max} - x_j^{\min}),\ j \in \overline{1,M}.$$

5. If $n < N$ then $n = n + 1$, go to a step 3.

Result is x^*.

2.6. Algorithm for parametric identification of the production audit data transformation model based on metaheuristics

For the proposed parametric identification method of the model for transforming production audit data based on metaheuristics, an algorithm has been developed that is designed to be implemented on a GPU using the CUDA information parallel processing technology and is shown in Fig. 3. This block diagram functions as follows.

Step 1. Operator input of the maximum number of iterations N, population size K, solution length M, maximum number of selected best solutions B, parameter to generate a vector of scale parameters β, $0 < \beta < 1$, minimum and maximum values for the solution $x_j^{\min}, x_j^{\max}$, $j \in \overline{1,M}$.

Step 2. Initialization of location parameters vector

$$\gamma^{loc} = (\gamma_1^{loc},...,\gamma_M^{loc}),\ \gamma_j^{loc} = x_j^{\min} + \frac{1}{2}(x_j^{\max} - x_j^{\min}).$$

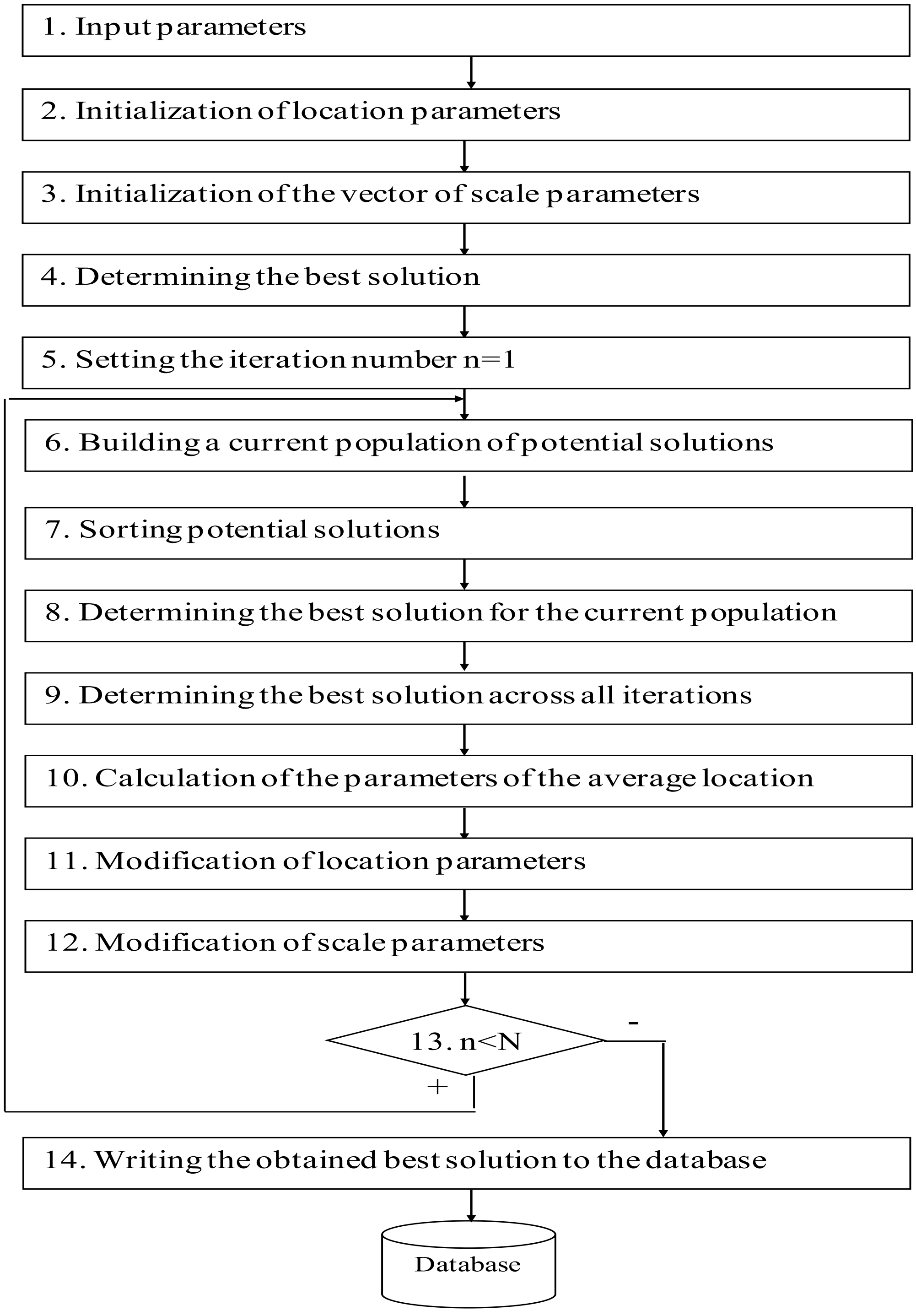

Figure 3: Block diagram of an algorithm for parametric identification of a model for transforming production audit data based on metaheuristics

Step 3. Initialization of the scale parameters vector

$$\gamma^{scale} = (\gamma_1^{scale}, ..., \gamma_M^{scale}),\ \gamma_j^{scale} = \beta(x_j^{\max} - x_j^{\min}).$$

Step 4. Determination of the best solution (the best vector of parameters of the MRMLP model)

$$x^* = \gamma^{loc}.$$

Step 5. Setting the iteration number $n = 1$.

Step 6. Building a current population of potential solutions P, those. generating each potential k-th solution using $K \cdot M$ GPU threads, which are grouped into K blocks. Each thread computes

$$x_{kj} = \gamma_j^{loc} + \gamma_j^{scale}\left(\left(\frac{N-n}{N}\right)C(0,1) + \left(\frac{n}{N}\right)N(0,1)\right).$$

Step 7. Sorting potential solutions by goal function, i.e. $F(x_k) < F(x_{k+1})$, based on parallel odd-even sort, using K GPU threads, which are grouped into 1 block.

Step 8. Determination based on parallel reduction of the best solution for the current population using K GPU threads, which are grouped into 1 block. Each thread computes a target function $F(x_k)$

$$k^* = \arg\min_{k \in \overline{1,K}} F(x_k).$$

Step 9. Determining the best solution across all iterations

if $F(x_{k^*}) < F(x^*)$, then $x^* = x_{k^*}$.

Step 10. Calculation of each j-th parameter of the average location $\tilde{\gamma}_j^{loc}$ based on parallel reduction using MB GPU threads, which are grouped into M blocks.

Step 11. Modification of each j-th location parameter using M GPU threads, which are grouped into 1 block. Each thread computes

$$\gamma_j^{loc} = \left(\frac{n}{N}\right)\gamma_j^{loc} + \left(\frac{N-n}{N}\right)\tilde{\gamma}_j^{loc}.$$

Step 12. Modification of each j-th scale parameter, using M GPU threads, which are grouped into 1 block. Each thread computes

$$\gamma_j^{scale} = \left(\frac{N-n}{N}\right)\beta(x_j^{\max} - x_j^{\min}).$$

Step 13. Termination condition.

If $n < N$, then $n = n + 1$ and go to step 6.

Step 14. Recording the best solutions for all iterations in the database.

3. Numerical research

The numerical research of the offered method of parametrical identification was conducted with use of technology of parallel processing of information of CUDA in a MATLAB package, at the same time the amount of threads in the unit corresponded to the population size, sorting of population was carried out on the basis of an algorithm of pair and unpaired sorting, search of the best solution on the current population x_{k^*} finding of an average vector of a location $\tilde{\gamma}^{loc}$ it was executed on the basis of an algorithm of a parallel reduction.

In this work population size $K = 3M$, maximum number of iterations $N = 100$, parameter for generation of a vector of parameters of scales $\beta = 0.1$, the maximum quantity of the selected best solutions $B = 0.1$K.

The results of the qualitative characteristics of the parametric identification methods of the proposed MRMLP neural network model, comparison in Table 1, where Q is the number of parameters for the MRMLP neural network, P is the power of the training set.

Table 1

Comparison of the qualitative characteristics of parametric identification methods of the proposed neural network model MRMLP

Parametric identification methods	coefficient of determination	computational complexity
Backpropagation method in sequential learning mode	0.80	~NPQ (forward / backward)
Proposed Metaheuristic Method Using GPU	0.95	~NK

4. Discussion

Backpropagation method in sequential mode:

- cannot be used in batch training mode, i.e. it is impossible to parallelize computations on a GPU, which reduces the speed of finding a solution (Table 1);
- high probability of hitting a local extremum, which reduces the accuracy of finding a solution (Table 1).

The proposed metaheuristic method eliminates the indicated disadvantages.

5. Conclusion

The article discusses the problem of reducing the risk of incorrect display of data sets in the audit DSS based on the method of neural network modeling of transformations of production audit data with recyclable waste and intermediates due to a modified recurrent multilayer perceptron (MRMLP). Reached further development a method of parametrical identification of the MRMLP model which considers number of iterations of training and combines Gaussian distributions and Cauchy that increases the forecast accuracy as on initial iterations all search space is investigated, and on final iterations the search becomes directed. The software implementing the offered methods in MATLAB package was developed and investigated on of the release of raw materials into production and the posting of finished products of a manufacturing enterprise with a two-year depth of sampling with daily time intervals. The made experiments confirmed operability of the developed software and allow to recommend it for use in practice in a subsystem of the automated analysis of DSS of audit for check of maps of sets of data of the raw materials release into production and the products output. Prospects of further research are in checking the offered methods on broader set of test databases.

References

[1] The World Bank, World Development Report 2016, Digital Dividends, 2016. URL: https://www.worldbank.org/en/publication/wdr2016

[2] A.V.Barmak, Y.V.Krak, E.A.Manziuk, V.S.Kasianiuk, Information technology separating hyperplanes synthesis for linear classifiers. Journal of Automation and Information Sciences, vol. 51(5), 2019, pp. 54-64. doi:10.1615/JAutomatInfScien.v51.i5.50

[3] T.V.Prokopenko, O.Grigor, Development of the comprehensive method to manage risks in projects related to information technologies, Eastern-European Journal of Enterprise Technologies vol. 2, 2018, pp. 37-43. doi:10.15587/1729-4061.2018.128140

[4] M.Schultz, M.Tropmann-Frick, Autoencoder Neural Networks versus External Auditors, Detecting Unusual Journal Entries in Financial Statement Audits, Proceedings of the 53rd Hawaii International Conference on System Sciences (HICSS-2020), Maui, Hawaii, USA, 2021, pp. 5421-5430. doi:10.24251/hicss.2020.666

[5] J.Nonnenmacher, F.Kruse, G.Schumann, G.Marx, Using Autoencoders for Data-Driven Analysis in Internal Auditing. In Proceedings of the 54th Hawaii International Conference on System Sciences, Maui, Hawaii, USA, 2021, pp. 5748-5757. doi:10.24251/hicss.2021.697

[6] Y.Bodyanskiy, O.Boiko, Y.Zaychenko, G.Hamidov, A.Zelikman, The Hybrid GMDH-Neo-fuzzy Neural Network in Forecasting Problems in Financial Sphere, Proceedings of 2nd International Conference on System Analysis & Intelligent Computing (SAIC), Kyiv, Ukraine, IEEE, 2020, pp. 1-6, doi:10.1109/SAIC51296.2020.9239152

[7] A.Sonika, M.Pratap; A.Chauhan, New technique for detecting fraudulent transactions using hybrid network consisting of full-counter propagation network and probabilistic network, 2016 International Conference on Computing, Communication and Automation (ICCCA), 29-30 April 2016, Greater Noida, India, IEEE. 2016, pp. 29-30. doi:10.1109/CCAA.2016.7813713

[8] T.Neskorodieva, E.Fedorov, I.Izonin, Forecast Method for Audit Data Analysis by Modified Liquid State Machine, Proceedings of the 1st International Workshop on Intelligent Information Technologies & Systems of Information Security (IntelITSIS 2020), Khmelnytskyi, Ukraine, 10-12 June, 2020, CEUR-WS, vol. 2623, 2020, pp. 25-35. URL: http://ceur-ws.org/Vol-2623/paper3.pdf

[9] T.Neskorodieva, E.Fedorov, Method for Automatic Analysis of Compliance of Expenses Data and the Enterprise Income by Neural Network Model of Forecast, Proceedings of the 2nd International Workshop on Modern Machine Learning Technologies and Data Science (MoMLeT&DS-2020), Lviv-Shatsk, Ukraine, 2-3 June, 2020: proceedings, CEUR-WS, Volume I: Main Conference, vol. 2631, 2020, pp. 145-158. URL: http://ceur-ws.org/Vol-2631/paper11.pdf

[10] I.Sutskever, J.Martens, G.E.Hinton, Generating text with recurrent neural networks, in Proceedings of the 28th International Conference on Machine Learning (ICML-11), 2011, pp.1017–1024

[11] T.Mikolov, M.Karafi´, L.Burget, J.Cernock'y, S.Khudanpur, Recurrent neural network-based language model, in Interspeech, vol. 2, 2010, p. 3

[12] M.Sundermeyer, et al., Translation modeling with bidirectional recurrent neural networks, Proceedings of the Conference on Empirical Methods on Natural Language Processing, October, 2014

[13] M.Berglund, T.Raiko, M.Honkala, L.Kärkkäinen, A.Vetek, J.T.Karhunen, Bidirectional Recurrent Neural Networks as Generative Models, In Advances in Neural Information Processing Systems, 28, Cortes C., Lawrence N.D., Lee D.D., Sugiyama M., Garnett R., Eds., Curran Associates, Inc., 2015, pp. 856−864
[14] M.Sundermeyer, R.Schluter, H.Ney, LSTM neural networks for language modeling. in Thirteenth Annual Conference of the International Speech Communication Association, 2012
[15] P.Potash, A.Romanov, A.Rumshisky, Ghostwriter: using an LSTM for automatic rap lyric generation, in Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, 2015, pp. 1919–1924
[16] E.Kiperwasser, Y.Goldber, Simple and Accurate Dependency Parsing Using Bidirectional LSTM Feature Representations, Transactions of the Association for Computational Linguistics, vol. 4, 2016, pp.313–327
[17] A.Graves, J.Schmidhuber, Framewise phoneme classification with bidirectional LSTM and other neural network architecture, Neural Networks, vol. 18, no. 5, 2005, pp. 602–610
[18] J.Chung, C.Gulcehre, K.Cho, Y.Bengio, Empirical evaluation of gated recurrent neural networks on sequence modeling, arXiv preprint arXiv: 1412.3555, 2014
[19] R.Dey, F.M.Salem, Gate-Variants of Gated Recurrent Unit (GRU) Neural Networks arXiv: 1701.05923, 2017
[20] H.Jaeger, M.Lukosevicius, D.Popovici, U.Siewert, Optimization and applications of echo state networks with leaky integrator neurons, Neural Networks, vol. 20, no. 3, 2007, pp. 335–352
[21] W.Maass, Liquid state machines: motivation, theory, and applications, In: Computability in context: computation and logic in the real world, 2010, pp. 275–296
[22] A.Nakib, El-G.Talbi, Metaheuristics for Medicine and Biology, Berlin: Springer-Verlag, 2017
[23] X.Yang, Nature-inspired algorithms and applied optimization, Charm: Springer, 2018
[24] S.Subbotin. A.Oliinyk, V.Levashenko, E.Zaitseva, Diagnostic Rule Mining Based on Artificial Immune System for a Case of Uneven Distribution of Classes in Sample, Communications, vol.3, 2016, pp. 3-11
[25] C.Blum, G.R.Raid, Hybrid Metaheuristics, Powerful Tools for Optimization, Charm: Springer, 2016
[26] X.-S.Yang, Optimization Techniques and Applications with Examples, Hoboken, New Jersey: Wiley & Sons, 2018. doi:10.1002/9781119490616
[27] R.Martí, P.M.Pardalos, M.G.C.Resende, Handbook of Heuristics, Charm: Springer, 2018. doi:10.1007/978-3-319-07124-4
[28] O.Bozorg-Haddad, M.Solgi, H.Loaiciga, Meta-heuristic and Evolutionary Algorithms for Engineering Optimization. Hoboken, New Jersey: Wiley & Sons, 2017. doi:10.1002/9781119387053.
[29] B.Chopard, M.Tomassini, An Introduction to Metaheuristics for Optimization. New York: Springer, 2018. doi:10.1007/978-3-319-93073-2.
[30] J.Radosavljević, Metaheuristic Optimization in Power Engineering, New York: The Institution of Engineering and Technology, 2018. doi:10.1049/pbpo131e.