Synthesis of the System of Iterative Dynamic Risk Assessment of Information Security

Denis Berestov[1], Oleg Kurchenko[1], Yuri Shcheblanin[1], Volodymyr Mishchenko[2], and Nataliia Mazur[3]

[1] *Taras Shevchenko National University of Kyiv, 24 Bohdan Hawrylyshyn str., Kyiv, 04116, Ukraine*
[2] *State Service of Special Communications and Information Protection of Ukraine, 13 Solomianska str., Kyiv, 03110, Ukraine*
[3] *Borys Grinchenko Kyiv University, 18/2 Bulvarno-Kudravska str., Kyiv, 04053, Ukraine*

Abstract

Information on the implementation of the architecture of the system of iterative dynamic risk assessment based on neural networks is given. The choice of network structure, types of neurons and algorithms for learning neural networks are substantiated.

Keywords

Risk assessment, information security, network, neural network.

1. Dynamic Iterative Risk Assessment as Part of a Continuous Audit System

To date, there is clearly a significant gap between the development of the technologies used in the creation of systems, and methods of assessment of protection effectiveness of these systems. The complexity of information systems is growing rapidly, which inevitably leads to an increase in the complexity of threat analysis and evaluation of its applied protection methods. Insufficient assessment of the created systems from the point of view of information security, in its turn, leads to emergence of new threats or rising probability of realization of old ones. Existing means of automation are poorly adapted to the development of technologies and do not allow for a comprehensive analysis of the relationship between the technologies used in terms of information security.In this regard, the issue of automating processes and tasks that are solved in the process of the analysis and risk management, as well as increasing the relevance of the results acquire greater importance. When using the classical approach, the time between the initial analysis of the system and the creation of a report often exceeds the time of emergence and implementation of threats. In addition, to carry out the procedure of information security risk analysis in the classical approach, it is necessary to conduct modeling of an automated system, which in itself is quite a difficult task.

Therefore, the following main drawbacks of the existing approaches to the assessment of information security risks can be distinguished: the complexity of work in conditions of obvious incomplete information about the components of risk and their ambiguous properties; the need to create a model of information system; the duration of the process and the rapid loss of relevance of the evaluation results; the complexity of aggregation of data from various sources, including statisticians and expert assessments; the need to involve individual specialists in risk analysis; subjectivity and ambiguity of the received estimations; difficulties in using assessments for management tasks and the complexity of process automation. In this regard, there is a need to obtain a gradually refined risk assessment in the course of the work of a specialist. By automating the process of accounting for threats associated with the emergence of new vulnerabilities in standard software and formalizing changes in the business landscape, you can create an environment that allows the specialist to create reports on the security of an information system, based on a series of consecutive reports for a short period of time. Processing

CPITS-II-2021: Cybersecurity Providing in Information and Telecommunication Systems, October 26, 2021, Kyiv, Ukraine
EMAIL: berestov@ukr.net (D. Berestov); kurol@ukr.net (O. Kurchenko); sheblanin@ukr.net (Y. Shcheblanin); mischen-ko_w@ukr.net (V. Mishchenko); n.mazur@kubg.edu.ua (N. Mazur)
ORCID: 0000-0002-3918-2978 (D. Berestov); 0000-0002-3507-2392 (O. Kurchenko); 0000-0002-3231-6750 (Y. Shcheblanin); 0000-0002-7578-1759 (V. Mishchenko); 0000-0001-7671-8287 (N. Mazur)

these data by using different methods of statistical forecasting will determine the optimal set of countermeasures taking into account “future risks” and thus raise the effectiveness of preventive countermeasures to significantly reduce the response time of the system to new vulnerabilities [1,2].

Returning to the definition, a continuous audit is an environment that allows the internal or external auditor to make judgments on significant issues, based on a series of reports created simultaneously or with a small interval. Accordingly, we define the concept of security risk analysis as an environment that allows a specialist to assess information security risks based on created - simultaneously or with a small interval - reports on the operation of the AS, means of protection of information and information security incidents related to the implementation of threats [3].

In order to implement a continuous security risk evaluation, it is necessary to create a system of dynamic iterative information security risk assessment. In this case by the input contour in a general form parameters of the system will be observed. In accordance with the approaches used in the work to obtain a risk assessment, it is necessary to assess the a posteriori probability of the realization of threats on the basis of observational data [4].

2. Architecture of the System of Dynamic Iterative Risk Assessment

The described approach served as a basis for the development of the architecture of the system of iterative dynamic risk assessment of information security. It is based on the principles of adaptability, universality and compatibility. By adaptability we mean the ability of the system to effectively perform specified functions in a wide range of changing conditions. In relation to the system of iterative dynamic risk assessment, the implementation of this principle is to use the mechanism of distributed collection, storage and processing of information as the main architectural solution.

The application of the principle of universality lies in the possibility of using the system for a wide range of tasks of information security risk analysis. The approach developed in this way can be used both in systems of intellectual content filtering and for creation of high-level reports which are used in the organizations for administrative decision-making.

The application of the principle of compatibility is that the complex should provide for the possibility of integration with the available in the automated system means of information protection and with the mechanisms of dynamic iterative risk analysis in the information security management system. Among other things, this approach also allows to quickly and efficiently implement new functionality in the complex, while maintaining a high level of reliability and stability of protective properties. Also the complex can be easily integrated with different systems and platforms. The diagram of data flows of the system is presented in Fig. 1.

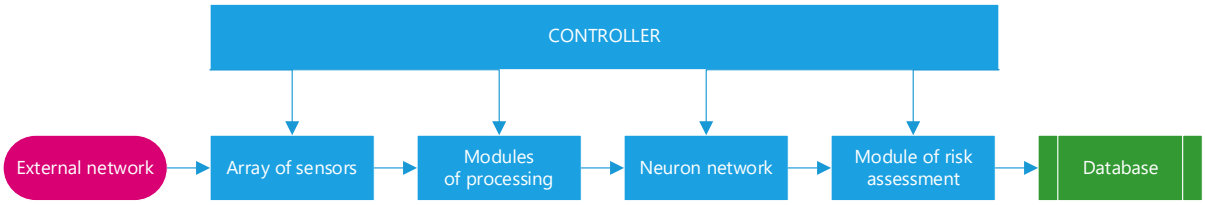

Figure 1: The diagram of data flows

The input of the system is sensor data (e.g., intrusion detection systems, anti-virus programs, firewalls) on potentially dangerous activity, the overall level of network activity and load on a particular part of the automated system, etc., as well as expert assessments of quantitative in-dicators of functioning of information security systems. From the array of sensors the data enters the risk assessment module. There the received data are transformed into a canonical form—for this purpose their normalization and alignment is carried out. Then, the processed data is fed to the input of the neural network. The neural network module solves the task of classification of input vectors and estimation of the a posteriori probability of belonging of input data to output classes. The results obtained during the probability assessment are included in the risk assessment module, where a direct quantitative assessment is performed. The obtained results are stored in a database for further use.

Classic client-server architecture was chosen as the architecture of the software package. In this case, given the specifics of the task, the agent's approach to data collection was applied. Security agents are installed at all key points of information exchange. The main functionality of security agents used to solve this problem is to collect and transmit the necessary data to servers of security management. In order to ensure the scalability of the architecture, security servers must provide the ability to create a hierarchy. In this case, the entire hierarchy of secu-rity servers implements the functionality of the risk assessment module and the function of the neural network. Distributed architecture allows one neural network to be physically imple-mented by multiple security servers if necessary.

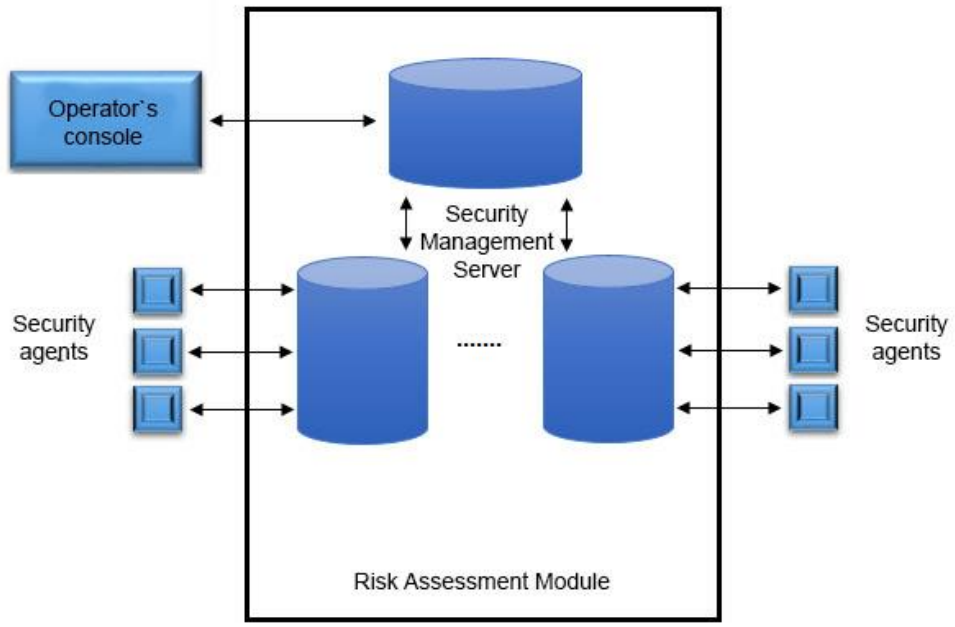

Figure 2: Scheme of interaction of modules of the complex

3. Research on the Application of Neural Networks

In mathematical statistics, the tasks of classification are also called the tasks of discriminant analysis. In machine learning the problem of classification is solved, as a rule, with the help of methods of artificial neural networks when setting up the experiment in the form of supervised training. Experimental methods of unsupervised training are used to solve another problem—clustering or taxonomy. In these tasks, the division of objects of the training sample into classes is not set, and it is necessary to classify objects only on the basis of their similarity with each other. In some applied areas, as well as in the mathematical statistics itself, due to the proximity of the tasks, clustering tasks are often no longer distinguished from classification tasks.

Mathematical problem statement is described as follows. Let X be a set of descriptions of objects, Y is a set of numbers (or names) of classes. There is an unknown target dependence—reflection $y^*: X \to Y$, the values of which are known only on the objects of the final training sample $X^m = \{(x_1,y_1), ..., (x_m,y_m)\}$. It is necessary to construct an algorithm: $X^m = \{(x_1,y_1), ..., (x_m,y_m)\}$, capable of classifying an arbitrary object $x \in X$.

Probabilistic statement of the problem is considered to be more general. It is assumed that the set of pairs “object, class” $X \times Y$ is a probability space with an unknown probability measure P. There is a final training sample of observations $X^m = -\{(x_1,y_1), ..., (x_m, y_m)\}$, generated according to probabilistic measure P. It is necessary to construct the algorithm: $X \to Y$ capable of classifying an arbitrary object $x \in X$.

A feature is a reflection of f: $X \to D_f$, where D_f is the set of admissible values of the feature. If the given features are $f_1,...,f_n$, then the vector $x = (f_1(x),\ldots,f_n(x))$ is called a characteristic description of the object xϵX. A characteristic description may be identified with the objects themselves. Thus the set $X = D_{f_1} \times ... \times D_{f_n}$ is called a space of features.

Depending on the set of D_f features are divided into the following types:

- Binary feature: $D_f = \{0,1\}$.
- Nominal feature: D_f is finite set.
- Ordinal feature: D_f is finite ordered set.
- Quantitative feature: D_f is the set of real numbers.

Often there are applied tasks with different types of features. Not all methods are suitable for their solution.

The following main types of classification tasks can be distinguished:

- Two-class classification.The most technically simple case, which is the basis for solving more complex problems.
- Multi-class classification. When the number of classes reaches many thousands (for example, in the recognition of hieroglyphs or merged speech), the task of classification becomes much more difficult.
- Non-perennial classes.
- Ordinary classes. An object can belong to several classes at the same time.
- Fuzzy classes. It is necessary to determine the degree of belonging of the object to each of the classes, usually a real number from 0 to 1 [5, 6].

During the information security audit, data on the parameters of the monitored system and expert assessment of control mechanisms are collected. On the basis of these observations, in fact, a conclusion is drawn about the affiliation of these input vectors of different classes of "danger" in terms of information security. In the simplest case, we can consider the following scenario: the collection of information about the activity in the AS takes place and on the basis of observations a conclusion is made about the danger of specific events or their safety. Such a task arises, for example, in the implementation of systems for detecting and preventing intrusions. In a more general case, the number of risk classes can be chosen arbitrarily. To solve the problem of classification it is necessary to construct the so-called discriminant function meant for the partition of an n-dimensional spatial vector into the desired classes. In most practical applications, it is necessary to find not the function itself, but its approximation.

The emergence of neural networks can be associated with the article by McCulloch and Pitts [7], which describes the mathematical model of the neuron and the neural network. It has been shown that both Boolean functions and finite state machines can be represented by neural networks.

Later, Rosenblnatt [8] proposed a model, which he called the perceptron, and a learning algorithm for such a model. It had been shown that perceptrons can solve some problems more efficiently than computers of traditional architecture. However, a serious mathematical analysis of perceptrons, conducted by Minski and Paperton [9], later revealed serious limitations in the applicability of perceptrons. In particular, it was shown that some problems, which in principle can be solved by a perceptron, might require unrealistically large amount of time for practical applications or unrealistically large number of neurons [10].

Later limitations were relaxed by replacing the threshold functions of neuronal activation with sigmoid. Tsybenko, Funahashi and Hornik independently proved the following fact. Let y be a fixed sigmoid function, and let f be a continuous function of n variables on compact $K \in R_n$. Then f can be approximated in the sense of a uniform approximation by a four-layer network (with two hidden layers), and the activation functions of the first and last layer are linear, and for the intermediate layers are equal to y.

Hecht-Nielsen proved the representability of the continuous function of multiple variables by means of a two-layer neural network with components of the input signal, $2n + 1$ components of the first (hidden) layer with sigmoid activation functions and M components of the second layer with unknown activation functions

Thus, in a nonconstructive form, the solvability of the problem of representing a function in a rather arbitrary form on a neural network was proved. These results have allowed the widespread use of ANN to solve many applied tasks, including information security risk analysis.

4. Architectures of Neural Network

New types of neural network architecture are constantly appearing, and they can be confusing. We have collected for you a kind of crib, which contains most of the existing types of ANN. Although all of them are presented as unique, the pictures show that many of them are very similar [11].

The architectures of neural networks can be divided into supervised and unsupervised networks that are taught with or without teachers. Mixed networks using both learning methods can be singled out. In the course of the analysis, various architectures of neural networks considered as the core of the system of dynamic iterative analysis of information security risks, examples of which are shown in Fig. 3.

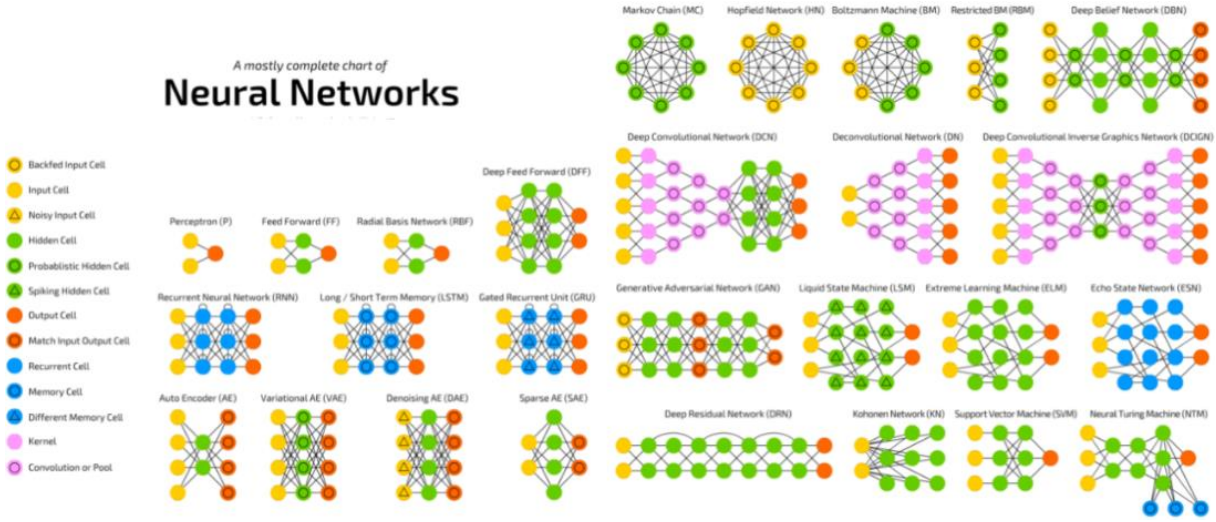

Figure 3: The example of the architecture of a neural network

In particular, we consider networks with learning algorithms of supervised and unsupervised learning, including multilayer perceptrons with direct signal propagation, counter-propagation networks based on Kohonen and Grossbeng neurons, self-organized Kohonen networks, Hopfield and Hemming feedback networks, dynamic associative memory networks, networks and algorithms based on adaptive resonance theory (ART), cognitrons and non- cognitrons. Comparisons of neural network architectures are given in Table 1.

To solve the problem of information security risk analysis, it is most appropriate to see the use of a multi-layered perceptron. In [12] it was shown that a multilayered perceptron learnt by the method of inverse error propagation can be used to estimate the a posterior probability. This issue, as well as the question of the optimality of such an assessment will be discussed in detail below. The use of a multilayer perceptron is the simplest option for using neural networks to solve the problem of information security risk assessment [13]. It can be shown that in the case of a smooth function with added Gaussian noise, a neural network with a coding scheme "1 with *c*" trained to minimize the standard deviation will approximate the a posteriori probability. However, to solve the problems of classification for the purpose of risk analysis, it is more relevant to analyze the solution for the problem of coding "1 with 2" from the distribution.

Table 1

Comparison of neural network architectures

Method of learningн	The rule of learningн	Architecture	Algorithm of learning	Task
Supervised	Correction of error	One-layer and multilayer perceptron	Algorithms of perceptron learning Reverse propagation Adaline i Madaline	Classification of images Approximation of function
	Boltzmann	Recurrent	Boltzmann algorithm of learning	Classification of images
	Hebb	Multilayer Direct propagation	Linear discriminant analysis	Data analysis Classification of images
	Competition	Competition	Vector quantization	Categorization of the inner class Data compression
		ART Network	ART Map	Classification of images
Unsupervised	Correction of error	Multilayer Direct propagation	Projection of Sammon	Categorization of the inner class Data analysis
	Hebb	Direct propagation or comparison	Analysis of main components	Data analysis Data compression
		Hopfield Network	Learning of associative memory	Associative memory
	Competition	Competition	Vector quantization	Categorization Data compression
		SOM of Kohonen	SOM of Kohonen	Categorization, data analysis
		ART Network	ART1, ART2	Categorization
Mixed	Correction of error	Network RBF	Algorithm of learning RBF	Classification of images Approximation of function Prediction Control

In this case, for the activation of the neuron network it is necessary to apply the sigmoid function of the activation of the form:

$$g(a) = \frac{1}{1 + e^{-a}}. \quad (1)$$

Then, given that all the values of the probabilities by definition lie in the range [0,1], the learning error function depends on the learning data, and has the form:

$$y = -\sum\sum_{k=1}^{c} \left\{ d_n^k \ln\left(\frac{y_k^n}{d_k^n}\right) + (1 - d_n^k) ln \frac{1 - y_k^n}{1 - d_k^n} \right\} \quad (2)$$

where y_k^n is orresponding output values, d_k^n is target values of training.

A well-known method of inverse error propagation can be used to teach such a multilayer perceptron. This method of learning a multilayer perceptron was first described by A. Galushkin and—independently and simultaneously—by P.J. Verbos. It was further substantially developed by D.I. Rumelhnarth, J. E. Hinton, and R. J. Williams as well as—both independently and simultaneously—by S.I. Kartsev and V.A. Okhonin. This is an iterative gradient algorithm that is used to minimize the error of the multilayer perceptron and obtain the desired output.

The idea of this method is to propagate error signals from the outputs of the network to its inputs, in the direction inverse to the direct propagation of signals in the normal mode. Bartz and Okhonin immediately proposed a general method (the “duality principle”) applicable to a broader class of systems, including systems with a delay, distributed systems, etc. For this method, there is proof of convergence, but it contains the assumption of an infinitely small step of adjusting the weights of neurons. In real conditions, the method does not always converge and has a number of drawbacks. P.D.

Wassermann described the adaptive algorithm of step selection, automatically correcting the step size in the learning process. The essence of the method is to combine the Cauchy machine with the idea of a gradient descent of reverse propagation that allows to construct a system that finds a global minimum while maintaining a high rate of reverse propagation. More sophisticated neural network architectures can also be used to address information security risk analysis. Their application is the direction of further research.

5. The Accuracy of the a Posteriori Probability Assessment with the Help of Multi-Layered Perceptron

Let's consider the task of classification once more. Let the space $X \in \mathfrak{H}^d$, which consists of objects, be divided into n groups. We denote the fact that the object x belongs to the j group as w_j. Let $P(w_j)$ be the probability of a randomly selected object belonging to the group j.

$f(x|w_j)$ is conditional density function of the probability that x will be a member of the group j of conditional probability

$$P(w_j|x) = \frac{f(x, w_j)}{f(x)}, \tag{3}$$

$$f(x, w_j) = f(x, w_j)P(w_j), \tag{4}$$

$$f(x) = \sum_{j=1}^{m} f(x, w_j). \tag{5}$$

Let us consider $\in \mathfrak{H}^d$ and $\in \mathfrak{H}^m$ as random variables, where $f_{x,y}(x,y)$ is function of joint probability distribution. For these conditions, we consider the problem of determining the reflection F: $\mathfrak{H}^d \to \mathfrak{H}^m$, such as:

$$Minimize\ E\left[\sum_{i=1}^{m}\left(y_i - F_i(x)\right)^2\right], \tag{6}$$

where E is this is the expected value defined for the co-distribution function, y_i and $F_i(x)$ is these are also components of $F_i(x)$, respectively. It is known that there is a solution to this problem:

$$F(x) = E[y|x]. \tag{7}$$

In the task of classification as x let us consider the same object and as y is variable class affiliation. For example, in this way:

$y_i = 1$, if the object x belongs to to the class I and $y_i = 0$ in other case, then

$$F_i(x) = E[y_i|x] = 1 \cdot P(y_i = 1|x) + 0 \cdot P(y_i = 0|x) = P(y_i = 1|x) = P(w_i|x). \tag{8}$$

So $F(x)$ *is* there is nothing but an a posteriori probability

Let

$$Q = E\left[\sum_{i=1}^{m}\left(y_i - F_i(x)\right)^2\right] = \int_{x\in X}\int_{y}[\sum_{i=1}^{m}\left(y_i - F_i(x)\right)^2]f_{x,y}(x,y)dxdy. \tag{9}$$

Using the dichotomous character of the variable y, joint distribution function $f_{XIy}(x,y)$ can be expressed as $f(x,w)$, where w is class affiliation vector.

In particular,

$$(x, y) = \begin{bmatrix}(x, w_1)\\(x, w_2)\\ \dots \\(x, w_m)\end{bmatrix}. \tag{10}$$

And the value Q, respectively, can be expressed as

$$Q = \int_{x\in X}\sum_{j=1}^{m}\left[\sum_{i=1}^{m}\left(y_i - F_i(x)\right)^2\right]f(x, w_j)dx = \tag{11}$$

$$= \sum_{j=1}^{m} \int_{x \in X} \left[\sum_{i=1}^{m} \left(y_i - F_i(x)\right)^2 \right] f(x, w_j) dx =$$

$$= \sum_{j=1}^{m} \int_{x \in X} \left[\left(y_i - F_i(x)\right)^2 + \sum_{i \neq j} F_i^2(x) \right] f(x, w_j) dx.$$

That, after transformations give us

$$Q = \int_{x \in X} \sum_{i=1}^{m} \left[F_i^2(x) - (2F_i(x) - 1) P(w_j|x) \right] f(x) dx = \tag{12}$$

$$= \int_{x \in X} \sum_{i=1}^{m} [(F_i(x) - P(w_i|x))^2 + P(w_i|x)(1 - P(w_i|x))] f(x) dx$$

Let's define

$$\sigma_A^2 = \int_{x \in X} \sum_{i=1}^{m} P(w_i|x)(1 - P(w_i|x) f(x) dx\,, \tag{13}$$

$$\sigma_\varepsilon^2 = \int_{x \in X} \sum_{i=1}^{m} (F_i(x) - P(w_i|x))^2 f(x) dx\,. \tag{14}$$

Then $Q = \sigma_A^2 + \sigma_\varepsilon^2$, σ_A can be interpreted as approximation error, and σ_ε is as the error of evaluation.

Now we consider a neural network with a direct propagation of the signal and the receiving object x at the input. We denote the connection of the neuron *i* and the neuron *j* as (*i,j*), and the weight of connection as w_{ij}. The output of the neuron we denote as $a_i = F_i(x_i)$, where F_i is activation function of the neuron w.

This network can be considered as a reflection F:$\mathfrak{H}^d \to \mathfrak{H}^m$, if *d* is dimension of the input vector, and *m* is the dimension of the output vector. The weights of the neural network are determined by the learning outcomes. Neural network learning can be seen as problematic error minimization, or

$$Minimize\ \frac{1}{L} \sum_{i=1}^{L} \sum_{i \in N_o} \left(y_i^l - a_i^l\right)^2, \tag{15}$$

where *L* is the number of learning vectors, and y_i^l is the target value (output value, according to the training element sample) in case of problematic classification y_i^l = 1, if the object belongs to the class and y_i^l = 0 if not.

A comparison of this expression with expression (8) shows that the problem of learning a neural network with direct signal propagation (multilayer perceptron) is the same least squares problem, where the expected value E is calculated on the learning set. Accordingly, of interest is the question of how accurate the estimate obtained by the neural network is.

We denote as l_m *m* the dimensional single hypercube $[0,1]^m$ and as $C(l_m)$ is the value range of the function on l_m.

The following theorem was proved by Tsybenko[14]:

Theorem. Let $\varphi(y)$ is a limited monotonically increasing real function other than a constant. Then for any function $f \in C(l_m)$ and for any $\varepsilon > 0$ there exists integer *N* and the corresponding set of constants a_i, $b_i \in R$, $w_i \in R^m$, where *i* = l,...,*N*, such that it is possible to determine the approximating function

$$F(x) = \sum_{i=1}^{N} \alpha_i\, \varphi(w_i^T x + b_i) \tag{16}$$

with property: $|F(x) - f(x)| < \varepsilon$ for all $x\ \epsilon\ l_m$.

Thus, the neural network, being a superposition of sigmoid functions can be considered as a universal approximator. It also follows from the above theorem that any function can be represented by an artificial neural network with one hidden layer. It can also be proved that any neural network with the number of hidden layers more than one, can be represented in the form of a neural network with one hidden layer

To fully describe the neural network, it is necessary to determine the number of neurons in the hidden layer. Determining the amount of neurons in the hidden layer in the general case is not an obvious task. The accuracy of approximation increases with the number of neurons in the latent layer. At H neurons the error of approximation is estimated as $O(1/H)$. However, an increase in the number of neurons in the hidden layer leads to the so-called retraining of the neural network [14]. Thus, the quality of information processing is significantly affected by the structure of the neural network. Insufficient or excessively large number of neurons in the hidden layer reduces the efficiency of information processing.

In solving practical problems, they usually resort to the empirical choice of network parameters. In the event that the conditions of the network are unknown, it is necessary to use the algorithm that ensures the selection of optimal parameters

In the works [14] for multilayer neural networks with serial connections and RBF networks, algorithms for structural-parametric optimization were obtained. These algorithms are based on cross-checking procedures, which represent one of the types of re-sampling, and are practically independent of a priori assumptions. Consistent use of a priori information in the training of neural networks allows Bayesian methodology, which represents a single conceptual basis for the formation of objective functions of both parametric and structural optimization. The development of constructive irreversible algorithms for finding the optimal structure of neural networks remains an urgent task.

6. Algorithm of Dynamic Iterative Risk Assessment of Information Security

We give a brief description of the logic of the algorithm for dynamic iterative risk assessment of information security using the proposed approach. The following is a description of the algorithm in the form of block diagrams. In Fig. 4 the basic block diagram of the algorithm of the system of iterative dynamic assessment of information security risks is given. When the algorithm is initialized, it is checked whether the neural network is trained or artificial. If the network is not trained, then it would be trained in accordance with the chosen algorithm. If the network is properly trained, then the input and preparation of data from sensors and data provided by technical experts are carried out.

At the next stage of the work of algorithm, the calculations of the a posteriori values of the probabilities of the realization of threats—that is, the output values of the network—are performed. Then the current data on the value of assets exposed to information security risks is obtained. At the final stage of the algorithm the calculation of the final risk assessment of information security is made. Then, the operation of the algorithm continues after a certain period of time, if no command is received to complete it.

Next, we consider the block diagrams of algorithms of network training and calculating its output values. Before this let’s look at a three-layered neural network with weights Wij, pictured in Fig. 4.

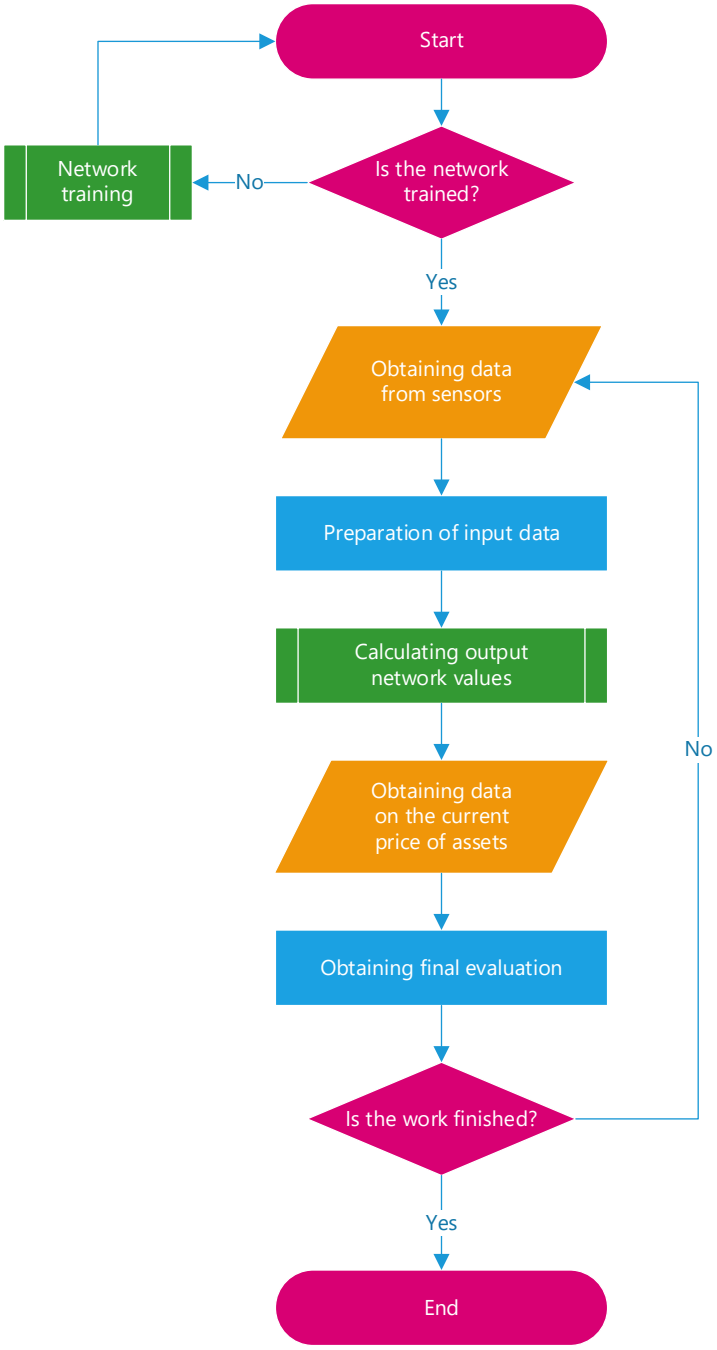

Figure 4: Block scheme of algorithm of itererative dynamic risk assessment

Let's denote as O_i the output of the i^{th} node. The purpose of the training is such adjustment of weights that at the output of the system a given input vector was obtained. It is assumed that for each input vector there is a corresponding output vector. Such a pair of vectors is called a training pair. A set of such pairs is required for training.

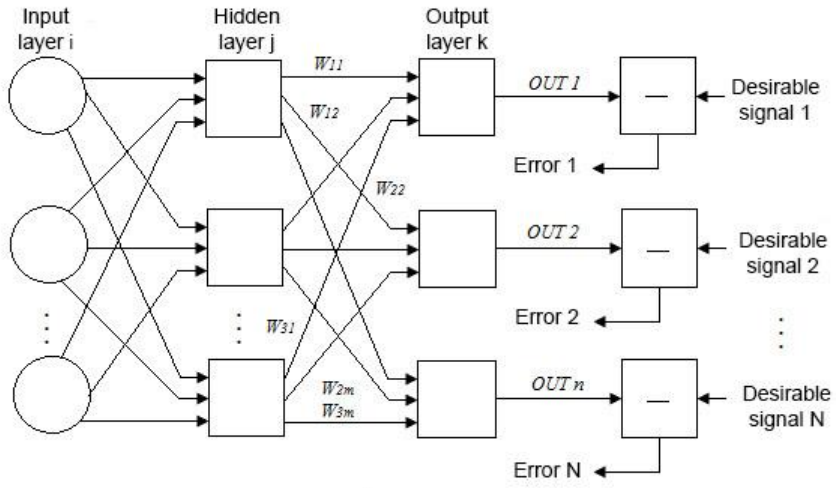

Figure 5: Scheme of multi-layered percetron (training)

Before the start of training, small initial values selected at random should be assigned to all weights. This choice ensures that the network will not be saturated with large values of weights and a number of other cases that can lead to errors.

Here is an algorithm for learning the reverse propagation network:

Step 1. Assign small random values of the weights.

Step 2. Choose a random pair of vectors from the learning set.

Step 3. Calculate the output of the neural network.

Step 4. Compare the initial output vector with the output training vector. Determine the value of the error. If this value is not acceptable, adjust the values of the weights and go to Step 3.

Step 5. Choose a new random pair of vectors not used in previous iterations. If there is such a pair, turn to Step 2.

Step 6. The end.

The block diagram of the algorithm for learning the reverse propagation network is presented in Fig. 6.

Adjustment of weights is carried out starting from the last layer. For the last level node:

$$\delta_j = -o_j(1 - o_j)(t_j - o_j) \tag{17}$$

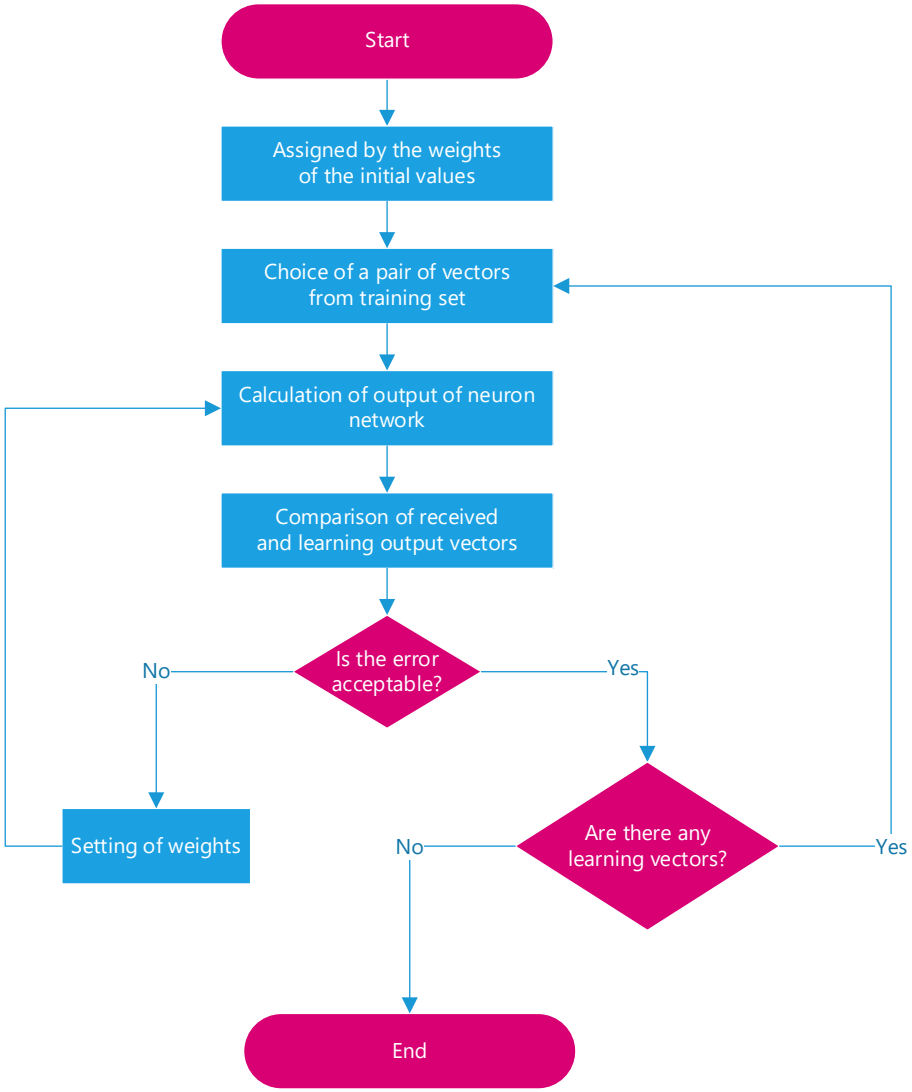

Figure 6: Block diagram of a learning algorithm of a reverse propagation network

For internal network node

$$\delta_j = -o_j(1-o_j)\sum_{k\in Outputs(j)}\delta_k w_{j,k} \tag{18}$$

where $0 < \eta < 1$ is a multiplier that specifies the speed of "movement." For each edge of the network $\{i,j\}$

$$\Delta w_{i,j} = \alpha\Delta w_{i,k} + (1-\alpha)\eta\delta_j o_j, \tag{19}$$

$$w_{i,j} = w_{i,j} + \Delta w_{i,j}. \tag{20}$$

Such adjustment is carried out for all layers with weights of connections.

7. Computational Complexity of the Algorithm of Dynamic Iterative Risk Assessment

Execution of the algorithm described above is necessary for all threats. Final complexity of the whole algorithm depends on the number of threats linearly, since for each threat a separate neural network is formed that does not depend on the type of threat. The calculating complexity of the algorithm for one threat is dependent on the complexity of the training of the neural network and a number of neurons in

it and does not depend on the input data. In the event that the network is trained, the calculating complexity is constant and depends linearly on the volume of input data.

Training of the network depends only on the volume of learning vectors and convergence of algorithm for the final time. That is, in the case of convergence of the algorithm the complexity of obtaining an assessment and complexity of learning linearly is dependent on the volume of input data. The calculating complexity T has the form:

$$T = O(n). \tag{21}$$

where n is the number of input vectors.

The convergence of the algorithm of reverse error propagation can be proved only for a finitely small step of change of weights. This leads to an endlessly large time of the work of algorithm. A method of adaptive choice of the step was proposed by P. D. Vasserman, which solves this problem and ensures the convergence of the algorithm to the global minimum for a finite number of steps. In addition, you can artificially set the number of iterations, which limits the time of the learning process.

4. Conclussions

The architecture of the system of iterative dynamic risk assessment with the use of Bayesian approach based on neural networks is described in detail. The choice of network structure, types of neurons and learning algorithms are substantiated.

Neural networks have the following properties:

- Allow directly to receive an assessment of the a posteriori value of probability.
- Can learn in the process of system operation.
- Ensure the automation of the process of accounting for threats and their assessment.
- Allow to provide aggregation of quantitative and qualitative data as well as automation of the process of obtaining assessments.
- Provide formalization of changes in the structure of the AS.
- Ensure the creation of assessments, which are gradually refined in the course of work with the system.

5. References

[1] Utkin L.V. http://www.levvu.narod.ru/Papers/Bayes.pdf

[2] Dreyfus, Neural networks: methodology and applications, Birkhauser, 2015.

[3] Berestov, D., et al., Analysis of features and prospects of application of dynamic iterative assessment of information security risks, in: Workshop on Cybersecurity Providing in Information and Telecommunication Systems (CPITS), vol. 2923, 329–335, 2021.

[4] Shevchenko, H., et al., Information security risk analysis SWOT, in: Workshop on Cybersecurity Providing in Information and Telecommunication Systems, CPITS, vol. 2923, 309–317, 2021.

[5] Lifshits Y. Automatic text classification (slides). Algorithms for the Internet.

[6] Merkov A. B. Basic methods used for handwriting recognition, 2015.

[7] McCulloch W., Pitts W. Logical calculus of ideas related to nervous activity. Avtomaty. Izd. foreign lit., 362–384, 1956.

[8] Rosenblatt F. Principles of neurodynamics. Perceptron and the theory of brain mechanisms. Mir, 1965.

[9] Minsky M., Papert S. Perceptrons, Mir, 1971.

[10] V. Buriachok, V. Sokolov, P. Skladannyi, Security rating metrics for distributed wireless systems, in: Workshop of the 8th International Conference on "Mathematics. Information Technologies. Education": Modern Machine Learning Technologies and Data Science (MoMLeT and DS), vol. 2386, 222–233, 2019.

[11] Barsky A.B. Neural networks: recognition, control, decision making. Finance and statistics, 2014. 176 p.

[12] GOST 19.701-90. Schemes of algorithms, programs, data and systems. Conventions and execution rules.

[13] Yakhyaeva G.E. Fuzzy sets, neural networks: textbook, 2nd ed., rev. Internet University of Information Technologies; Binomial. Knowledge Lab, 2010.

[14] Milov V.R. Training of neural RBF-networks based on structural-parametric optimization procedures. Neurocomputers: development and application, 5, 29–33, 2003.