EasyChair Preprint

№ 6590

Applying of Machine Learning for Analyzing Network Traffic in the Conditions of an Unbalanced Data Sample

Babyr Rzayev and Ilya Lebedev

September 13, 2021

Applying of machine learning for analyzing network traffic in the conditions of an unbalanced data sample

Babyr Rzayev[1][0000−0002−9671−650X] and Ilya Lebedev[2][0000−0001−6753−2181]

[1] S.Seifullin Kazakh Agrotechnical University, 010000 Nur-Sultan, Republic of Kazakhstan
pathinchaos@gmail.com
[2] St. Petersburg Federal Research Center of the Russian Academy of Sciences, 199178 St.Petersburg, Russian Federation
isl_box@mail.ru

Abstract. The article provides a solution to the problem of identifying anomalous situations in information and telecommunication systems, based on artificial intelligence methods. The presented method for identifying an anomalous situation is based on processing the received tuples of network traffic packet values using various classification models. The proposed solution improves the identification accuracy and makes it possible to use classification algorithms optimized for different types of events and anomalies, trained on various training samples, presented in the form of tuples of network packet parameters. The difference between the algorithms is achieved by introducing an imbalance in the training samples. The paper describes the experiment using Naïve Bayes, Hoeffding Tree, J48, Random Forest, Random Tree, REP Tree machine learning classification algorithms, and the Multilayer Perceptron neural network. The method can be applied in information security monitoring systems when analyzing network traffic. A feature of the proposed solution is the possibility of its scaling and combination by adding new algorithms for classification of machine learning.

Keywords: Anomaly detection · Network traffic · Information security.

1 Introduction

The functioning of information and telecommunication systems (ITS) requires continuous monitoring of the emergence of various failures and collisions related to a network traffic processing. The development of the Industrial Internet and the Industrial Internet of Things (IoT) concepts causes the need to assess the performance, functional safety of individual network devices and network segments formed by them. The status analysis is carried out using various monitors processing internal and external information containing statistical data of network packets and their processing indicators. As a result, multidimensional time series and tuples of values appear that contain a time-variable parameters reflecting the functioning of the system.

The growth and the constant accumulation of network traffic information in various segments of Industrial networks and IoT necessitates the use of artificial intelligence methods for processing time series and tuples of values in order to detect anomalies.

Depending on the tasks of analyzing the status of information security of communication channels and the ITS itself, a number of directions based on classification, clustering and forecasting can be distinguished [1–4].

The implementation of such approaches causes certain difficulties. During the operation of the IoT devices, conflicts may occur both at the level of the information system and a separate network segment or device that influence the processes of receiving and transmitting information, increase the loading of communication channels, reduce the performance and processing speed of commands and messages [5–7]. Detection and preventing such situations requires improving models, methods for monitoring the state, allowing to analyze causal relationships and transitions. Adaptation of methods of statistical analysis, the formation of precedent, event models makes it possible to forecast to prevent negative consequences [5–10].

The detection of an anomalous situation in a network segment is based on statistical information of traffic in various modes and states and is carried out using neural networks, Markov models, machine learning methods, and others. The formed tuples of features have the form of patterns intended for analyzing the behavior of the device in various modes of operation [11–14].

The main disadvantage of existing approaches is that the systems that implement them may not always be effective in the face of constant changes in configuration and architecture. In this regard, it becomes necessary to develop and adapt methods for analyzing information that are resistant to changing operating conditions, providing a given completeness and accuracy of detecting anomalous situations.

The purpose of this work is to improve the quality indicators of event identification in network traffic by using an ensemble of algorithms trained on various unbalanced training samples. The proposed solution is aimed at achieving a variety of classifiers. The resulting effect of the spread of the algorithms' answers is smoothed out by applying the voting procedure.

2 The proposed approach

Modern network traffic analysis in most cases is based on machine learning methods. Automatic, without the participation of an expert, feature extraction and versatility are undoubtedly positive qualities of such approaches. However, due to the specifics of the field of information security, attacks on devices, networks and telecommunications used by an attacker each time have a certain "novelty" and uniqueness. On the other hand, the share of secure connections (P2P services, HTTPS), where traffic is compressed and encrypted, is increasing. Consequently, problematic situations may arise associated with the functioning of algorithms, input parameters, the extraction of analyzed features and their interpretation

for various kinds of information influences and attacks. In this regard, a method for identifying an anomalous situation in an information and telecommunication system is proposed, based on bagging, which ideally makes it possible to analyze the destructive effects on the ITS by comparing the results of several classifiers.

A formalized description of the construction of an ensemble of classifying algorithms is presented in [11, 15–17].

In general, the problem looks like this. There is a sample where the values of the studied parameters are obtained for various states, which allows us to associate a set of tuples from the set X with each state.

$X_i = (x_1, ..., x_n)$ contains a tuple of $n \geq 2$ length values.

The set of states is defined by tuples $\{X_1, X_2, \ldots, X_m\} \in X$ where m is the number of records in the sample, reflecting the behavior of the process in various states. Each state is assigned a label of a binary set of a subset of dangerous C_1 and safe C_2 states.

Thus, there is a labeled final training set. It is necessary for the input tuple of values X_i to construct a classification algorithm $a = \{b_1, b_2, ..., b_k\}$, mapping $X \to C$, where k is the number of basic classifying algorithms in the ensemble.

During the operation of the classification algorithm, an error occurs, which can be smoothed out by a sequence of independently trained basic classifiers $b_q, q = 1, ..., k$ [17, 18].

The response of the algorithm $b_q(X_i)$, obtained as a result of the work, allows us to determine the class of subsets $C_j, j = 1, 2$, belonging to the binary set of classes of dangerous and safe states C.

The result of the operation of the algorithms $b_q, q = 1, ..., k$ separately is the probability that the input tuple X_i belongs to the class of states:

$$b_q(X_i) = max_{j=1,2} P_q(C_j|X_i). \tag{1}$$

The resulting class predicted by the ensemble of classifiers for a tuple X_i can be determined based on the values of the functions F_1 and F_2 for the binary subset C_1 and C_2:

$$F_1(b_1(X_i), ..., b_k(X_i)) = 1/k \sum_{q=1}^{k} P_q(C_1|X_i), \tag{2}$$

$$F_2(b_1(X_i), ..., b_k(X_i)) = 1/k \sum_{q=1}^{k} P_q(C_2|X_i). \tag{3}$$

The implementation of the ensemble of basic algorithms is described by the expression:

$$b(X_i) = f(F_1(b_1(X_i), ..., b_k(X_i)), F_2(b_1(X_i), ..., b_k(X_i))), \tag{4}$$

where f is the decision rule that allows you to determine the probabilistic estimate and establish the class number.

Thus, the given solution uses a binary classification. The formalization is summarized by a decision rule that translates the grade into a class number. To

calculate the results of the ensemble under consideration, an auxiliary set (space of estimates) is used.

In order to achieve the difference between the algorithms included in the model, their training occurs independently of each other, both on randomly selected and unbalanced subsets of the training sample. The work examines the quality indicators of the ensemble when an unbalanced sample is used.

3 Experimental study of an ensemble of classifiers

Experimental evaluation of the considered approach was carried out on the NSL-KDD dataset which having 41 attributes and one more attribute for class. The dataset contains more than 30 types of attacks. These attacks can be divided in four different types with some common properties. These categories attacks include Denial of Service (DoS), Probe, Remote to Local (R2L), User to Root (U2R) attacks. Detailed description of the data set is presented in the works [18, 19].

As part of the experiment, a binary classification of the states of a telecommunication system (identification of parasitic and normal traffic) was carried out.

One of the problematic issues was the composition of the ensemble of classifiers. There are only some recommended rules related to the formation of training samples. Most of the research is devoted to the use of "weak" classifying algorithms in the ensemble. In a number of works, in order to ensure conditions aimed at achieving the "difference" of classifiers, it is recommended not to use training subsamples that are resistant to change [20, 21]. The use of exact classifiers in conjunction with relatively weak classifiers [22] is considered, combinations of ensembles, in which there are algorithms unequal in terms of quality indicators, are investigated [22–25]. Approaches using stacking, blending, multilevel stacking are presented [18, 21, 22].

To date, there are no regulated rules for forming the composition of an ensemble, which requires additional research and experiments. The choice of the model is based on the problem, where the composition of the algorithms is determined in advance, and its change will be difficult.

In the case of functioning in an autonomous mode, depending on the conditions, arising anomalies, training opportunities and the formation of a training sample, classifiers can be both strong and weak. In this regard, it is necessary to investigate the characteristics in different situations.

The formation of a training sample is an important component for the development of a successful model [18, 26, 27]. There are certain recommendations for the formation of training sets, but in practice their implementation is difficult. It is not always possible to obtain the probability of the appearance of objects and signs equal to the probability of their appearance in the general population. Problematic issues in the formation of training samples are associated with the detection, separation and correct interpretation of background and significant patterns, the absence of training objects of a certain type and elements of the

feature system, inaccurate ranges of values of variables, imbalance, the appearance of external patterns associated with the conditions for the formation of a training set [26].

In the experiment, the case of voting was considered, when both strong and weak classifiers participated in the ensemble, the diversity of which was formed by the training sample. The assessment was carried out for classifiers: Naïve Bayes (NB), Hoeffding Tree (HT), J48, Random Forest (RF), Random Tree (RT), REP Tree (REP). The analysis results were obtained using the free Weka software.

The sample was marked up and divided into two parts, one of which was used as training, and the other was used for testing. The data structure vector included more than 40 attribute values, which are described in [19].

The resulting dataset of the network traffic is considered as a sequence of tuples of values (1) of heterogeneous packet, about 50% of which reflects the normal traffic and 50% is the parasitic.

The set was divided in proportions 20/80, 40/60, 60/40 and 80/20, where the upper part of the proportion shows the value of the sample for training, and the lower part for testing.

The classification algorithms were trained using the standard Weka settings. The correspondence of the tuples of the tested sample to one of the classes of dangerous or safe states was determined by expressions (2), (3) and (4).

The training samples were formed artificially. Initially, it was assumed that the structure is unbalanced; for each classifier, individual events are presented with different frequencies.

Figure 1 shows an example of sample modification, when each classifier was given its own “competence area”.

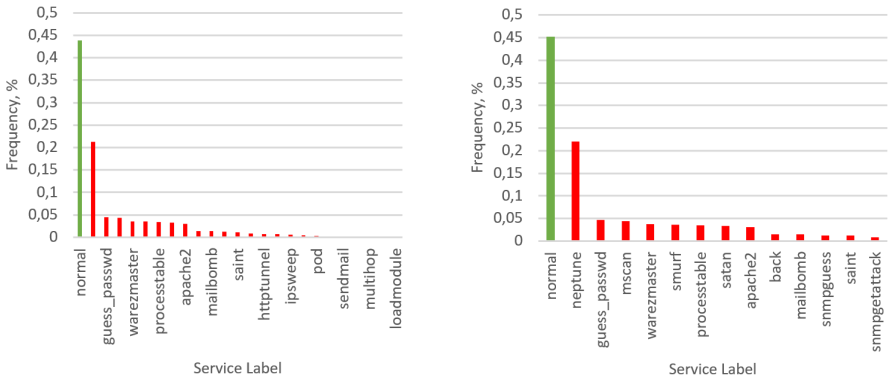

Fig. 1. Sample modification. (a) is a whole sample, (b) is an example of an unbalanced sample for a single classifier.

A typical situation was simulated when the classifier can show good results on the training set, but the quality indicators fall when working with test (in the simulated case) or real data, i.e., unaccounted features appear and the training set does not fully reflect the general population.

In the first part of the experiment, classification was carried out using machine learning algorithms - NB, HT, J48, RF, RT, REP. Assessment of classifiers was carried out on the basis of the area under the ROC-curve (AUC) [26] for the test set (Figure 2).

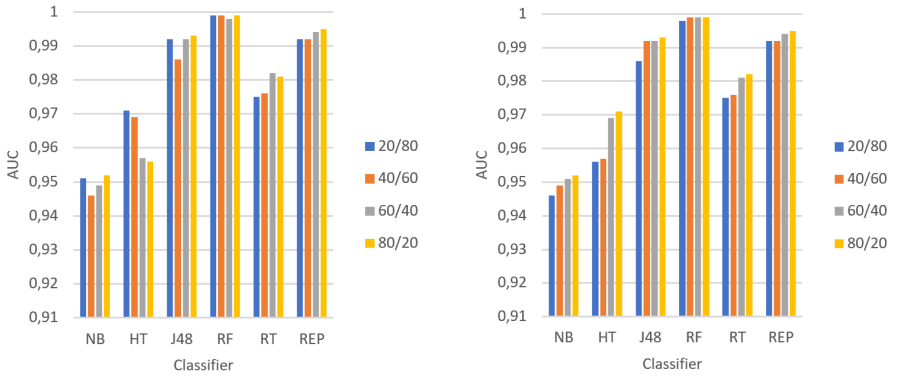

Fig. 2. Assessment of the quality of classifiers by the area under the ROC-curve with balanced (a) and unbalanced (b) training sample in ratios 20/80, 40/60, 60/40, 80/20.

The figure shows the sample size of experimental data from the dataset and the quality of algorithms for binary classification. The histogram (figure, b) makes it possible to determine the "weak" and "strong" classifiers that arose after the use of unbalanced training samples as a result of incomplete features, incorrect determination of dependencies, etc. The classifiers were assessed based on the values of the Accuracy, Precision, Recall, F-Measure parameters. The obtained experimental values are presented in Figure 3.

In the second part of the experiment, an ensemble of classifiers is implemented (3). The incoming data were simultaneously processed by all algorithms, and on the basis of expressions (2) and (3), the class of the subset was determined, which was compared with the pre-labeled test set.

For the analysis of the ensemble of classifiers, AUC and an indicator of the overall accuracy of the sequence of classifiers were selected. The results obtained after using the ensemble are shown in Figure 4 (a). The third part of the experiment uses the Multilayer Perceptron (MLP) neural network. The results are shown in Figure 4 (b).

A comparative table for the AUC parameter for all classifiers is shown in Figure 5.

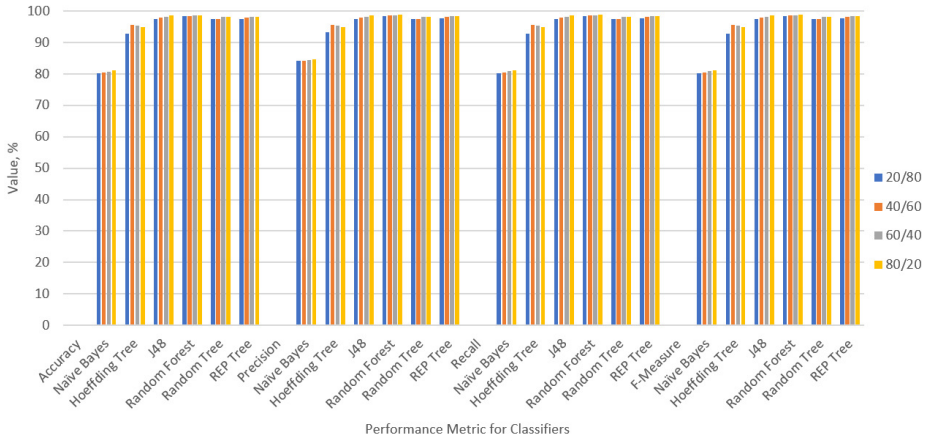

Fig. 3. Accuracy, Precision, Recall and F-Measure for Individual Classifiers.

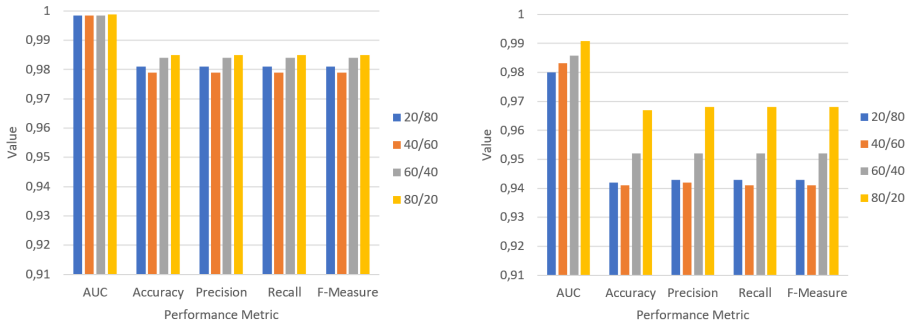

Fig. 4. Results of applying Bagging of Classifiers (a) and MLP (b) in ratios 20/80, 40/60, 60/40, 80/20.

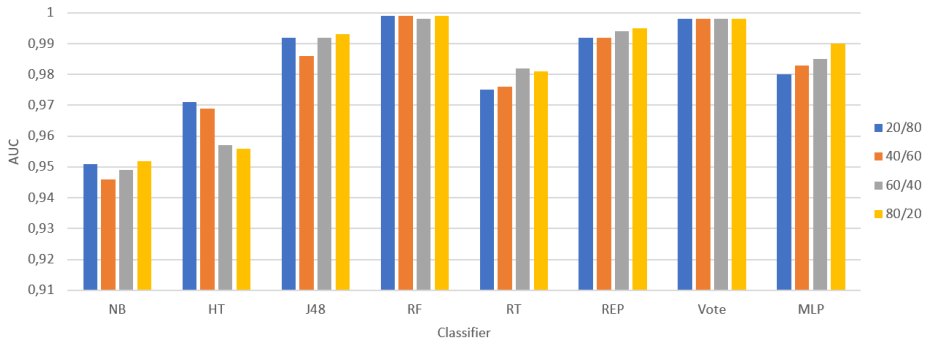

Fig. 5. Assessment of the quality of classifiers by the area under the ROC-curve in ratios 20/80, 40/60, 60/40, 80/20.

The results of testing the open NSL-KDD dataset with machine learning classifiers implemented in the Weka application, using bagging, showed an accuracy of more than 99%, even on a relatively small training set.

The main advantage of the proposed solution is that a relatively small sample is sufficient to achieve results comparable with other methods [19, 20, 23]. In contrast to traditional approaches to the formation of a training set for algorithms, imbalanced training samples are created, which makes it possible to achieve a "variety" of classifiers [18, 21, 25, 26]. The classification algorithms are adjusted for different types of events and anomalies, and the resulting effect of the scatter of responses is smoothed by an ensemble. Each classifier "specializes" on a certain part of events, which allows it to be adapted to different conditions of network segments functioning.

Among the disadvantages of the proposed approach, it is necessary to note the sensitivity of the classification algorithms to the bias of responses. It is necessary to analyze the data, feature space and classifiers in advance for the possibility of this effect. In the case of its strong influence, the results of the ensemble can be significantly reduced.

4 Conclusion

The growing volume of network traffic necessitates the development of models, methods of its analysis in order to identify destructive influences and anomalous situations. The number of new types of attacks, malicious sites, and methods of introducing unauthorized software is constantly increasing. In this regard, it is necessary to analyze a large number of parameters of network information exchange.

One of the main indicators of monitoring systems is the accuracy of state identification. The paper proposes a method based on bagging classifiers to identify anomalous situations in network traffic. Taking into account the large number of processed indicators, the proposed approach allows obtaining results acceptable in terms of accuracy, smoothing out potential errors of heterogeneous classifiers. The variety of classifiers is achieved by using unbalanced training samples. When using the ensemble, as the volume of the training sample increases, an increase in accuracy is observed.

The method has a limitation associated with the possible manifestation of the effect of displacement of responses by classifying algorithms. In this regard, before use, there is a need for additional research of data and classifiers.

The advantage is the ability to scale and combine it by adding new classifying algorithms, taking into account the parameters of network traffic in various network segments.

References

1. Ahlgren B., Hidell M., Ngai E.: Internet of things for smart cities: interoperability and open data. IEEE Internet Computing, 2016, vol. 20, no. 6, pp. 52–56. https://doi.org/10.1109/MIC.2016.124

2. Salehi H., Burgueño R.: Emerging artificial intelligence methods in structural engineering. Engineering Structures, 2018, vol. 171, pp. 170–189. https://doi.org/10.1016/j.engstruct.2018.05.084
3. Oikarinen E., Tiittanen H., Henelius A.: Detecting virtual concept drift of regressors without ground truth values. Data Mining and Knowledge Discovery, 2021, vol. 35, Issue 3, pp. 821-859. https://doi.org/10.1007/s10618-021-00739-7
4. Gokhale A., McDonals M.P., Drager S., McKeever W.: A Cyber Physical System Perspective on the Real Time and Reliable Dissemination of Information in Intelligent Transportation Systems. Network Protocols and Algorithms, 2010, vol. 2. no. 3, pp. 116–136. https://doi.org/10.5296/npa.v2i3.480
5. Maletzke A., dos Reis D., Batista G.: Combining instance selection and self-training to improve data stream quantification. Journal of the Brazilian Computer Society, 2018, vol. 24, no. 12, pp. 123-141. https://doi.org/10.1186/s13173-018-0076-0
6. Kwon D.W., Ko K., Vannucci M., Reddy A.L.N., Kim S.: Wavelet methods for the detection of anomalies and their application to network traffic analysis. Quality and Reliability Engineering International, 2006, vol. 22, no. 8, pp. 953–969. https://doi.org/10.1002/qre.781
7. Semenov V.V., Lebedev I.S., Sukhoparov M.E.: Approach to classification of the information security state of elements for cyberphysical systems by applying side electromagnetic radiation. Scientific and Technical Journal of Information Technologies, Mechanics and Optics, 2018, vol. 18, no. 1, pp. 98–105. (in Russian). https://doi.org/10.17586/2226-1494-2018-18-1-98-105
8. V. López, A. Fernandez, S. Garcia, V.: Palade and F. Herrera An Insight into Classification with Imbalanced Data: Empirical Results and Current Trends on Using Data Intrinsic Characteristics. Information Sciences, 2013, vol. 250, no. 7, pp.113-141. https://doi.org/10.1016/j.ins.2013.07.007
9. Genkin D., Shamir A., Tromer E.: Acoustic cryptanalysis. Journal of Cryptology, 2017, vol. 30, no. 2, pp. 392–443. https://doi.org/10.1007/s00145-015-9224-2
10. Semenov V.V., Lebedev I.S., Sukhoparov M.E., Salakhutdinova K.I.: Application of an Autonomous Object Behavior Model to Classify the Cybersecurity State. Internet of Things, Smart Spaces, and Next Generation Networks and Systems, 2019, pp. 104–112. https://doi.org/10.1007/978-3-030-30859-9_9
11. Palacios A., Sanchez L., Couso I.: Combining Adaboost with preprocessing algorithms for extracting fuzzy rules from low quality data in possibly imbalanced datasets. International Journal of Uncertainty Fuzzines and Knowledge-based Systems, 2012, vol. 20, no. 2, pp. 51–71. https://doi.org/10.1142/S0218488512400156
12. Sethi T., Kantardzic M.: Handling adversarial concept drift in streaming data. Expert Systems with Applications, 2018, vol. 97, pp. 18–40. https://doi.org/10.1016 / j.eswa.2017.12.022
13. Jin J., Gubbi J., Marusic S., Palaniswami M.: An information framework for creating a smart city through internet of things. IEEE Internet of Things Journal, 2014, vol. 1, no. 2, pp. 112–121. https://doi.org/10.1109/JIOT.2013.2296516
14. Sukhoparov M.E., Semenov V.V., Salakhutdinova K.I., Lebedev I.S.: Identification of anomalous functioning of Industry 4.0 devices based on behavioral patterns. Information Security Problems. Computer Systems, 2020, no. 1, pp. 96–102. (in Russian)
15. Semenov V., Lebedev I., Sukhoparov M.: Identification of the state of individual elements of cyber-physical systems based on external behavioral characteristics. Journal of Applied Informatics, 2018, vol. 13, no. 5(77), pp. 72–83. (in Russian)

16. Sukhoparov M.E., Lebedev I.S.: Identification the information security status for the internet of things devices in information and telecommunication systems. Systems of Control, Communication and Security, 2020, no. 3, pp. 252–268. (in Russian). https://doi.org/10.24411/2410-9916-2020-10310
17. Rzayev B. T., Lebedev I. S.: Applying bagging in finding network traffic anomalies. Scientific and Technical Journal of Information Technologies, Mechanics and Optics, 2021, vol. 21, no 2, pp. 234-240 (in Russian). https://doi.org/10.17586/2226–1494-2021-21-2-234-240
18. Ingre B., Yadav A.: Performance Analysis of NSL-KDD dataset using ANN. Proc. 4th International Conference on Signal Processing and Communication Engineering Systems (SPACES), 2015, pp. 92–96. https://doi.org/10.1109/SPACES.2015.7058223
19. Dhanabal L., Shantharajah Dr. S.P.: A Study on NSL-KDD Dataset for Intrusion Detection System Based on Classification Algorithms. International Journal of Advanced Research in Computer and Communication Engineering, 2015, vol. 4, no. 6, pp. 446–452. https://doi.org/10.17148/IJARCCE.2015.4696
20. Vorontcov K.V.: Lectures on algorithmic compositions. http://www.machinelearning.ru/wiki/images/0/0d/Voron-MLCompositions.pdf. Last accessed 11 May 2021
21. D'yakonov A.G.: Solution methods for classification problems with categorical attributes. Computational Mathematics and Modeling, 2015, vol. 26, no. 3, pp. 408–428. https://doi.org/10.1007/s10598-015-9281-2
22. Zhou Z.-H.: Ensemble Methods: Foundations and Algorithms. New York, CRC Press, 2012, 222 p.
23. Zhou Z.-H. and Feng J.: Deep Forest. National Science Review, 2019, vol. 6, no. 1, pp. 74–86.https://arxiv.org/abs/1702.08835v4. Last accessed 20 May 2021
24. Khan S., Yairi T.: A review on the application of deep learning in system health management. Mechanical Systems and Signal Processing, 2018, vol. 107, no. 1, pp. 241–265. https://doi.org/10.1016/j.ymssp.2017.11.024
25. Pedersen T.: A simple approach to building ensembles of naive bayesian classifiers for word sense disambiguation. NAACL 2000: Proc. of the 1st North American chapter of the Association for Computational Linguistics Conference, 2000, pp. 63–69
26. Kaftannikov I.L., Parasich A.V.: Problems of training set's formation in machine learning tasks. Bulletin of the South Ural State University. Series Computer Technology, Aotimatic Control, Radio Electronics, 2016, vol. 16, no. 3, pp. 15–24. (in Russian). https://doi.org/10.14529/ ctcr160302
27. Fawcett T.: An introduction to ROC analysis. Pattern Recognition Letters, 2006, vol. 27, no. 8, pp. 861–874. https://doi.org/10.1016/j.patrec.2005.10.010