Vol: 2, Issue: 1, 2024
ISSN: 2980-3152
Pages:1-12

CURRENT TRENDS IN COMPUTING
https://dergipark.org.tr/en/pub/ctc
RESEARCH ARTICLE

AUTOENCODER-BASED INTRUSION DETECTION IN CRITICAL INFRASTRUCTURES

HAKAN CAN ALTUNAY[1], ZAFER ALBAYRAK[2*] AND MUHAMMET ÇAKMAK[3]

[1] *Department of Computer Technologies, Carsamba Chamber of Commerce Vocational School, Ondokuz Mayis University, Türkiye*
[2] *Department of Computer Engineering, Faculty of Technology, Sakarya University of Applied Sciences, Türkiye*
[3] *Department of Computer Engineering, Faculty of Engineering and Architecture, Sinop University, Türkiye*

ABSTRACT. **Securing critical infrastructure systems such as electricity, energy, health, management, transportation, and production facilities against cyber attacks is the issue on which states spend the most time and money when creating security strategies. Every day, different methods have emerged to prevent attackers who endanger our personal and national security with varying types of attacks. The most important of these methods is intrusion detection systems. This study proposes an autoencoder-based intrusion detection system model to detect security anomalies in critical infrastructures. The accuracy of this proposed model in attack detection has been tested with the current and complex UNSW-NB15 dataset. In the proposed model, training and testing steps were carried out using the attack packages in the data set. These packages are then divided into two: normal or attack. According to the results obtained in the experiments, it has been confirmed that the proposed intrusion detection system can effectively detect attacks with high performance.**

1. INTRODUCTION

Modern societies depend on advanced cyber and physical infrastructures to carry out daily activities [1]. These infrastructures are also called critical assets that protect services not only in the physical world but also in the digital world. Today, the protection of these infrastructures, which cover different areas such as communication, transportation, and energy, has become a national security concern [2]. Ensuring the continuity, control, and security of the services provided by critical infrastructures is a costly and difficult process [3].

Cyber attacks are carried out against SCADA systems in various areas, such as nuclear power plants, electrical networks, and water treatment plants [4]. Figure 1 shows the major attacks on industrial control systems around the world since 2010.

E-mail address: zaferalbayrak@subu.edu.tr (*).
Key words and phrases. Intrusion Detection System, Critical Infrastructure, Autoencoder.

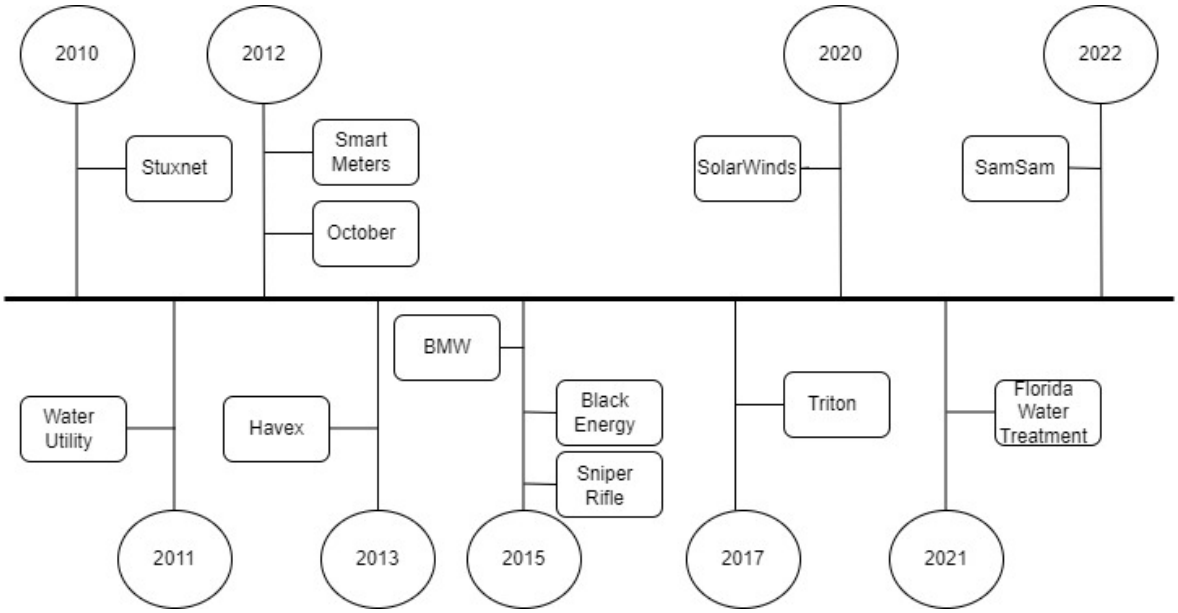

FIGURE 1. **Major attacks on critical infrastructures.**

It is a fact that the use of the Internet in our daily lives increases information sharing, interpersonal communication, and interaction [5]. The Internet of Things (IoT), defined as the intelligent connection of objects that communicate by sensing each other, is frequently used today. With IoT, data communication on devices in the network can be monitored, and this data can be collected using sensors with a wireless network connection [6].

Intrusion detection systems (IDS) are preferred to prevent cyber attacks and reduce their effects. Security is provided by IDS in the transmission of data on the network from the source to the receiving node. Therefore, IDS plays an important role in ensuring network security. Machine learning is a sub-branch of artificial intelligence [7]. With its ability and capacity to improve, it can empower various systems to learn from experience and make decisions without any explicit programming. Machine learning approaches are generally divided into supervised and unsupervised [8]. Classification of traditional IDSs is generally presented as signature-based, anomaly-based, and hybrid IDSs. Signature-based IDSs extract behavioral patterns of intruders [9]. IDSs are generally classified as anomaly-based, signature-based, and hybrid-based. IDSs in which the behavior of attackers trying to enter the network without permission is kept as signatures are called signature-based IDSs [10]. These signatures are compared with the attack types that the network has encountered before [11]. If the signatures match as a result of this comparison, the packet is detected as an attack. This type of IDS does not produce false positive values. They detect any intrusion with a signature pattern. In these IDSs, attacks with unknown signatures cannot be detected, and accordingly, a high rate of false negative values is produced [12]. If the database has an updateable feature, the false negative value rate can be reduced. IDSs that analyze events in the network

are anomaly-based IDSs. In this type of IDS, normal states and abnormal states are distinguished [13]. In anomaly-based IDSs, the behavioral profiles of users in the system are first determined. Behaviors that differ from normal behaviors are defined as abnormal behaviors. The higher the correct detection rate of normal behavior profiles, the higher the correct detection rate of abnormal behaviors. In anomaly-based IDSs, normal behaviors are continuously updated [14]. Some attack types cannot be detected by anomaly-based intrusion detection systems. Therefore, anomaly-based systems may have a high false positive rate. In hybrid-based systems, signature and anomaly-based systems are used together [15]. In this way, a much more reliable network and management system emerges [16].

Considering the above problems, the main subject of this study is the design of an autoencoder-based intrusion detection system to detect attack packets by detecting intrusions with high performance in critical infrastructure systems where the types of attacks and the amount of data increase day by day. The proposed model subjects intrusion packets to binary classification.

The primary motivation and contribution of this study are summarized below. The number of cyber attacks on critical infrastructures is increasing day by day. These attacks cause material and moral losses. Therefore, it is important to detect these attacks and protect the system. In this study, an intrusion detection system for critical infrastructures is implemented using an autoencoder.

Secondly, the proposed model for intrusion detection systems was tested on UNSW-NB15, a complex dataset with a large amount of data and a high number of attributes, taking into account the increasing amount of data in critical infrastructures. The performance of the proposed model was evaluated using the binary classification procedure carried out on the current data set, providing a more reliable and observable process.

2. Related Work

Davis and Clark suggested an in-depth examination of request packets to increase the performance of anomaly-based IDSs developed with the machine learning method and to detect increasing types of attacks and emphasized that data pre-processing has a great impact on the success of anomaly-based IDSs [17].

In their study, Naseer and Saleem tried various categorical data coding methods, chose the most appropriate method for the data set they used, and performed hyperparameter optimization by using the random search method in the models established with the Deep Convolutional Neural Network (DCNN) algorithm. It has been stated that pre-processing and hyper-parameter optimization significantly improve the attack detection rate and speed of the created models [18]. In another study, Hancock and Khoshgoftaar emphasized that stable categorical data coding techniques are suitable for large data sets due to their low running time and low computational complexity [19]. Tang et al. reached an accuracy rate of 89.82% in the data set on which they applied categorical data coding with the one hot encoding method and feature selection pre-processing with the Light Gradient Boosting Machine (LightGBM) algorithm and the attack detection model they created with the Autoencoder (AE) algorithm [20].

Aslan et al. analyzed the malware behavior in the system and proposed a Subtractive Central Behavior Model to detect this malware. In the proposed model, attributes were created by analyzing malware

behaviors and the system in which the behaviors were performed. Additionally, the obtained features were reduced by proposing a new feature selection algorithm. With the proposed model, 99.9%, 0.2%, and 99.8% rates were achieved in detection rate, false positive rate, and accuracy metrics, respectively [21].

Mazini et al. applied hyperparameter optimization to the data set after categorical data coding and scaling pre-processing and made feature selection with the artificial bee colony algorithm. The resulting data set and AdaBoost. In the model created with the M2 classifier, 99.61% detection, 0.01% false detection, and 98.90% correct detection rates were achieved [22]. By selecting features according to the information gain rate, Balakrishnan et al. achieved 99.11% detection success and 2.08s detection time in Denial of Service (DoS) attacks with the data set with the resulting feature subset and the Support Vector Machine (SVM) classification algorithm [23].

Torabi et al. mentioned the importance of using different and up-to-date data sets to prove the generalization success of the developed intrusion detection models [24]. Ozkan Okay et al. proposed a hybrid attack detection model and achieved 99.65% and 99.17% accuracy rates with KDD99 and UNSW-NB15 datasets, which were pre-processed with a feature selection approach (FSAP) algorithm [25].

Ambusaidi et al. achieved 98.90% attack detection success and 0.521% false positive rate with the data sets on which they applied hybrid feature selection using mutual information (MI) and helical sequential forward selection (SFS) methods [26].

Chen et al. have used datasets consisting of different combinations and intersections of features selected by Principal Component Analysis (PCA), C4.5, and Genetic Algorithm (GA) techniques, achieved the most successful results with the features selected jointly by PCA and GA techniques [27].

Song mentioned that since traditional feature selection algorithms are insufficient for variable-size datasets, this problem can be solved with online feature selection algorithms [28].

Kanimozhi and Jacob performed hyperparameter optimization for the number of hidden layers and alpha parameters using the grid search method in the anomaly-based intrusion detection model they created using the Multilayer Perceptrons (MLP) algorithm, and achieved 99.97% accuracy, 0.001% false positive and 99% F-criterion rates [29].

In their study, Latah and Toker used the NSL-KDD dataset for anomaly-based attack detection in software-defined networks (SDN), 12 different classifiers, and the PCA approach for feature extraction from the dataset. As a result of the experiments, the model established with the Decision Tree (DT) algorithm showed the best performance in precision, AUC, F1-measure, McNemar, and accuracy metrics. While bagging and boosting approaches outperform other traditional machine learning methods such as Extreme Learning Machines (ELM), K Nearest Neighbors (KNN), Random Forest (RF), Neural Networks (NN), Latent Dirichlet Allocation (LDA), and SVM with a 99.5% confidence level, the best results were achieved in FAR and recall metrics with LogitBoost. The best test time was obtained with ELM [30].

Uğurlu et al. In their study, they selected 30 attributes through the weighting process from 82 attributes in the CICDarknet2020 data set, which they used to detect and classify darknet traffic. In the study, the grid method was used for hyper-parameter adjustment, and as a result of the experimental studies, an accuracy rate of 93.32% was achieved with the Decision Tree algorithm [31].

3. DATASET

Using the UNSW-NB15 dataset IXIA PerfectStorm tool, a hybrid model was created in Australian cyber security center laboratories, including both real modern normal activity and artificial network traffic attack movements suitable for today's conditions [32].

The developers of the dataset also divided the dataset into two different groups: the training dataset and the testing dataset. This data set was later used by many researchers. The training data set consists of 175341 records, and the test data set consists of 82332 records. The original data set consists of 2540044 records [33]. In this study, the subsample data set, which was created by the developers of the original data set and divided into training and test data sets, which many researchers use in their studies, was used as the data set. The data set used does not contain any unnecessary records. The UNSW-NB15 dataset has a total of 49 features and one target value. The value distribution of the types of attacks in the UNSW NB15 dataset is presented in Table 1.

TABLE 1. **Distribution of attack values in the UNSW - NB15 dataset**

Attack Types	Number
Fuzzers	18184
Backdoor	1746
Analysis	2000
General	40000
Shellcode	1133
Reconnaissance	10491
DoS	12264
Worms	130
Exploits	33393
Benign	56000

In recent years, IDS has been increasing performance using deep learning methods at points where existing traditional security solutions are insufficient. In particular, anomaly-based IDSs play a very important role in detecting attacks known as zero-day attacks. One of the most important factors to evaluate the performance of IDSs and to create more effective and efficient IDSs is the data sets used [32]. The data set used must comply with the age requirements and include current attack types. The UNSW-NB15 data set, which is frequently preferred in the literature, meets modern conditions in a variety of attack types and normal traffic scenarios, and the regular distribution of training and test data sets are the positive aspects of this data set [33].

Deep autoencoders, a specific application of artificial neural networks, are used to perform unsupervised learning. In the deep autoencoder, the data is first compressed and encoded. Then, a representation closest to the input data is obtained from the code whose features are reduced. [34].

The autoencoder learns how to remove noise from the data to reduce data sizes. An autoencoder consists of 3 parts: encoder, code, and decoder. The encoder is where the input data is compressed. In this section, the code is generated. The decoder reconstructs the input data using this code. In other words, an autoencoder cannot be created without an encoder, decoder, and loss function. The loss function is

used in the autoencoder to compare the output with the targeted result [35]. Figure 2 shows a general deep autoencoder architecture.

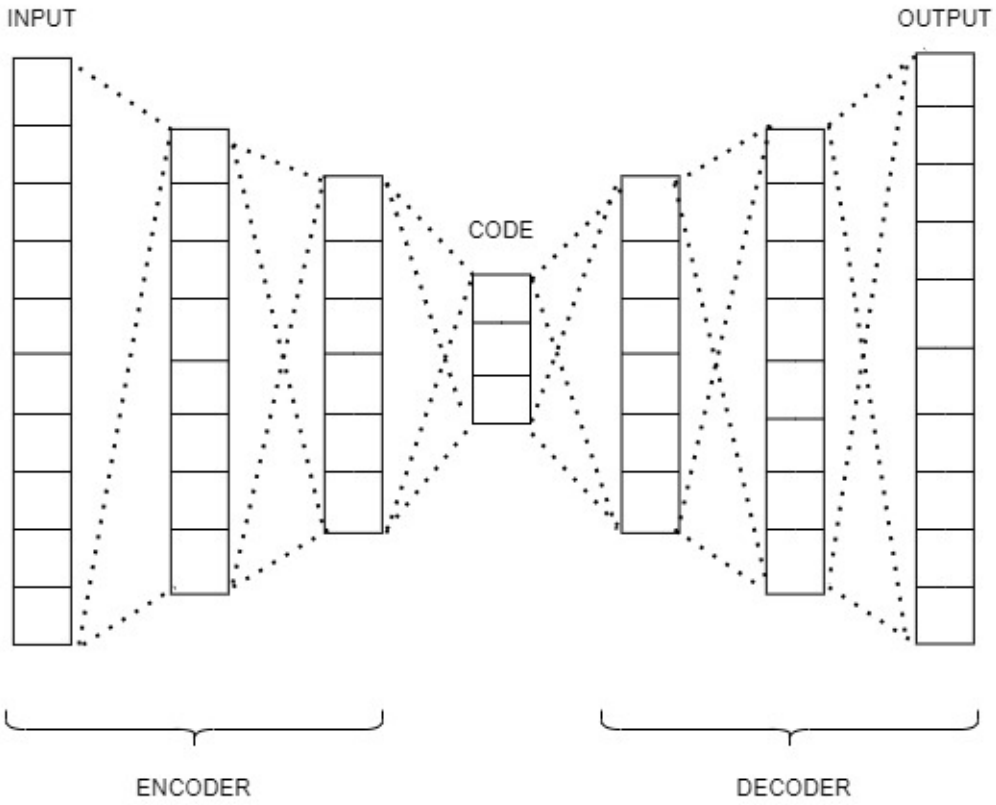

FIGURE 2. **Deep autoencoder architecture.**

The equations of the encoder and decoder sections are shown below.

$$Y = f_{\emptyset}(X) = s(WX + b_x) \tag{1}$$

$$X^{'} = g_{\emptyset'}(Y) = s(W^{'}Y + b_Y) \tag{2}$$

During the autoencoder training process, reconstruction loss is minimized in the dataset. The objective function is used here. The following equation is used to determine the parameters that will minimize the loss value along with the objective function.

$$\emptyset = minL(X, X^{'}) = minL(X, g((f(X)))) \tag{3}$$

Autoencoder is the deep learning model used in this study. Recall, precision, F1-Score, and accuracy were used for evaluation criteria as in [36]. The following equations were used to obtain the relevant metrics.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{4}$$

$$Recall = \frac{TP}{TP+FN} \tag{5}$$

$$Precision = \frac{TP}{TP+FP} \tag{6}$$

$$F1-Score = \frac{2*Precision*Recall}{Precision+Recall} \tag{7}$$

First, data from the training and test sections of the data set were taken. These data were then applied to the Autoencoder model. At the output of the Autoencoder model, the data is classified as attack or normal. Figure 3 shows the architecture of the proposed model.

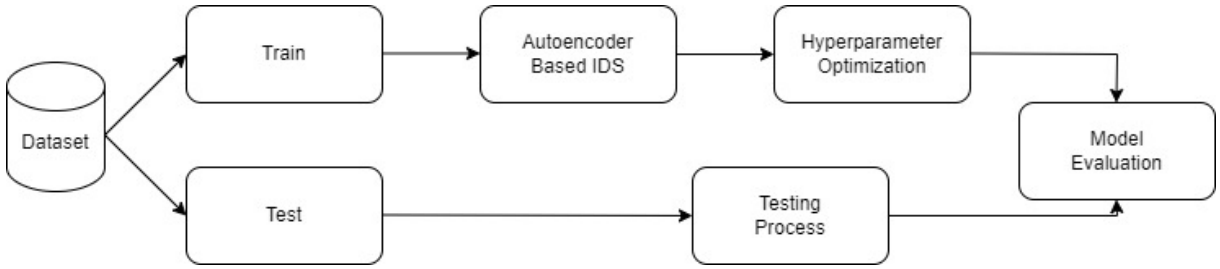

FIGURE 3. **Architecture of the proposed model.**

The experimental study divided the data set into training and testing to prevent overfitting. One of the critical issues of machine learning is the generalization of the algorithms or models we have developed. Generalization is the ability to observe how well the model works with the data you have learned and with new, previously unseen data we will obtain in the future. We can briefly define it as getting good results with the latest data.

Our primary and greatest goal in machine learning studies is to create a model that accurately predicts previously unknown data elements. Therefore, the created learning model must be generalized very well to ensure the accurate classification of future data items. Generalization means our model is good at learning from given data and applying the learned information elsewhere. If it performs well on data it has not seen in training, it generalizes well on the provided data [37]. This study's data set is divided into 80% training and 20% testing.

The multi-layered architectural structures that come with deep learning have brought a series of hyper-parameter groups waiting to be decided by the designer. Some of these parameters are used to select

the basic algorithm to be applied in the model from several algorithm groups, such as the optimization algorithm and activation function. Since the number of algorithms is limited, it is generally relatively easy to select such hyperparameters.

However, the number of layers, neurons, learning coefficients, kernel size, etc. Hyperparameter types also expect us to choose from a set that extends to infinity within certain limits or on the number line. The selection of such hyperparameters is a laborious and time-consuming process. Our first choices regarding hyperparameters when designing a model do not yield the right results. By changing the hyperparameters one after the other iteratively, the model's performance is observed, and the most appropriate hyperparameter group for the model is selected. In addition, some methods automate this selection process.

In this study, the heuristic parameter fitting method was used for hyperparameter optimization. In this method, hyperparameters are estimated using our prior knowledge of the problem, the model is designed according to these hyperparameters, and the results are observed. According to the results, the model is rebuilt and trained by making new hyperparameter estimates that will intuitively increase the model's performance, and the results are observed. This process continues until suitable parameter groups that will give the expected performance are found [38]. Hyperparameters of the model used in the study are shown in Table 2.

TABLE 2. **Hyperparameters of the proposed model**

Hyperparameters	**Value**
Input Neurons	45
Hidden Neurons	32
Output Neurons	45
Iteration	650

4. RESULTS AND DISCUSSION

The study used the autoencoder model with original data without making a feature selection in performance evaluation. Without feature selection, the training dataset and testing dataset were used separately. The results obtained are shown in Table 3 and Figure 4.

According to these results, the proposed autoencoder-based intrusion detection system reached 97.63% accuracy in determining attack packets. The accuracy value obtained was supported by precision, recall, and F1 Score values. Hyperparameter optimization was carried out in the tests carried out with the UNSW-NB15 data set, which is very rich in terms of the number and diversity of attack packages. The ROC curve obtained as a result of the study is shown in Figure 5.

The accuracy rates obtained in studies using different data sets in the literature and the rates obtained in this study are shown in Table 4. The results obtained show that the autoencoder method has a high detection accuracy in attack classification.

TABLE 3. **Performance metrics and results**

Metrics	Value
Accuracy	97.63%
Precision	97.14%
Recall	97.78%
F1 - Score	97.45%

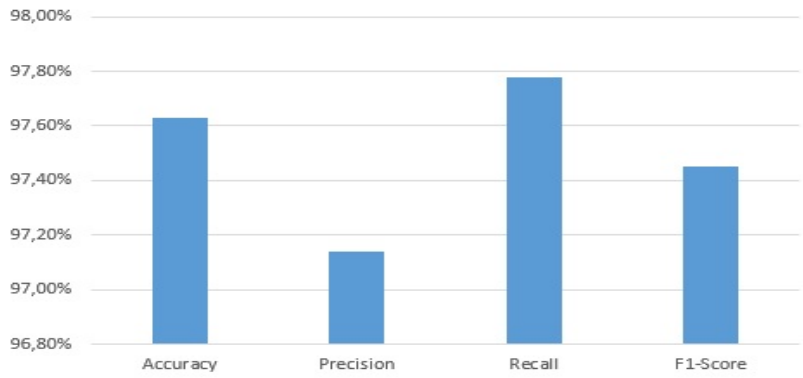

FIGURE 4. **Results obtained in the experimental study.**

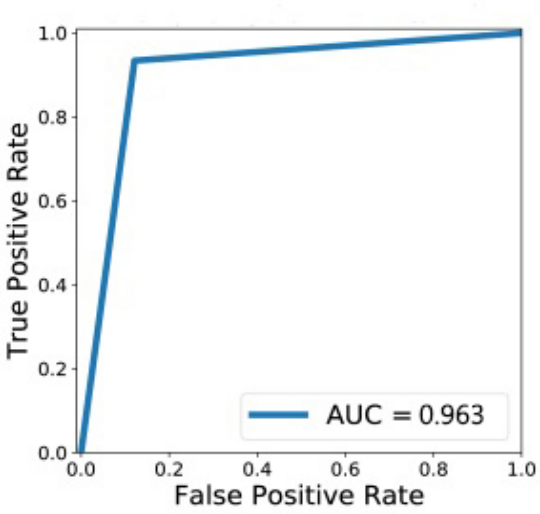

FIGURE 5. **ROC of proposed model.**

TABLE 4. **Comparison of the proposed model with studies in the literature**

Reference	Model	Accuracy
[18]	DCNN	85.22%
[20]	LightGBM + Autoencoder	89.82%
[21]	SCBM + J48	99.8%
[22]	Artificial Bee Colony + AdaBoost	98.90%
[31]	Decision Tree	93.22%
Proposed model	Autoencoder	97.63%

5. CONCLUSION

In our study, an autoencoder-based intrusion detection system is proposed. In this system, which can detect abnormal behavior in the network with high performance, no feature extraction is made from the data set. According to the results, the proposed autoencoder model reached a 97.63% accuracy value. In addition, 97.14% precision, 97.78% recall, and 97.45% F1-Score values were achieved in the model. It is seen that this study achieves higher performance compared to other studies in the literature. The main reason for this situation is the use of an up-to-date data set and hyperparameter optimization. It is planned to prepare algorithms based on feature selection in the future. In this way, the effect of feature selection on classification accuracy will be investigated. In addition, future studies need to examine the detection time of attack symptoms. Considering that the number of institutions and organizations with critical infrastructure is increasing day by day, it is thought that deep learning-based intrusion detection systems will be needed, especially in this field.

DECLARATIONS

- **Conflict of interest:** The authors have not disclosed any competing interests.
- **Data availability:** The data will be shared upon request.

REFERENCES

[1] Kasongo S.M., Sun Y., Performance Analysis of Intrusion Detection Systems Using a Feature Selection Method on the UNSW-NB15 Dataset, J Big Data, 7, 105, 2020.

[2] Yadav M.S., Kalpana R., Data Preprocessing for Intrusion Detection System Using Encoding and Normalization Approaches, 2019 11th International Conference on Advanced Computing (ICoAC), ChennaiIndia, 265-269, 18-20 Aralık, 2019.

[3] Liu H., Zhou M., Liu Q., An embedded feature selection method for imbalanced data classification, in IEEE/CAA Journal of Automatica Sinica, 6 (3), 703-715, 2019.

[4] Alabadi, M., Habbal, A., Wei, X., Industrial internet of things: Requirements, architecture, challenges, and future research directions, IEEE Access, 2022.

[5] Alaca, Y.,Çelik, Y., Cyber attack detection with QR code images using lightweight deep learning models. Computers & Security, 126, 103065, 2023.

[6] Kutluana, G., Turker, I., Classification of cardiac disorders using weighted visibility graph features from ECG signals, Biomedical Signal Processing and Control, 87, 105420, 2024.
[7] Altunay, H. C., Kritik Altyapılara Yönelik Derin Öğrenme Tabanlı Saldırı Tespit Sistemi Tasarımı, (Doctoral dissertation), 2023.
[8] Altunay, H., C., Albayrak, Z., Network Intrusion Detection Approach Based on Convolutional Neural Network, Avrupa Bilim ve Teknoloji Dergisi, (26), 22-29, 2021.
[9] Bharadiya, J. P., Machine learning and AI in business intelligence: Trends and opportunities, International Journal of Computer (IJC), 48(1), 123-134, 2023.
[10] Sharifani, K., Amini, M., Machine Learning and Deep Learning: A Review of Methods and Applications, World Information Technology and Engineering Journal, 10(07), 3897-3904, 2023.
[11] Choi, S., Yoon, S., Energy signature-based clustering using open data for urban building energy analysis toward carbon neutrality: A case study on electricity change under COVID-19, Sustainable Cities and Society, 92, 104471, 2023.
[12] Landauer, M., Wurzenberger, M., Skopik, F., Hotwagner, W., Höld, G., Aminer: A modular log data analysis pipeline for anomaly-based intrusion detection, Digital Threats: Research and Practice, 4(1), 1-16,2023.
[13] Bhavsar, M., Roy, K., Kelly, J., Olusola, O., Anomaly-based intrusion detection system for IoT application, Discover Internet of Things, 3(1), 5, 2023.
[14] Sharma, B., Sharma, L., Lal, C., Roy, S., Anomaly based network intrusion detection for IoT attacks using deep learning technique, Computers and Electrical Engineering, 107, 108626, 2023.
[15] Hnamte, V., Hussain, J., DCNNBiLSTM: An efficient hybrid deep learning-based intrusion detection system, Telematics and Informatics Reports, 10, 100053, 2023.
[16] Yin, Y., Jang-Jaccard, J., Xu, W., Singh, A., Zhu, J., Sabrina, F., Kwak, J., IGRF-RFE: a hybrid feature selection method for MLP-based network intrusion detection on UNSW-NB15 dataset, Journal of Big Data, 10(1), 1-26, 2023.
[17] Davis J.J., Clark A.J., Data preprocessing for anomaly based network intrusion detection: A review, Computers & Security, 30 (6-7), 353- 375, 2011.
[18] Naseer S., Saleem Y., Enhanced Network Intrusion Detection Using Deep Convolutional Neural Networks, KSII Trans. Internet Inf. Syst, 12 (10), 5159-5178, 2018.
[19] Hancock J.T., Khoshgoftaar T.M., Survey on categorical data for neural networks, Journal of Big Data, 7, 1-41, 2020.
[20] Tang C., Luktarhan N., Zhao Y., An Efficient Intrusion Detection Method Based on LightGBM and Autoencoder. Symmetry, 12 (9), 1458, 2020.
[21] Aslan, Ö., Samet, R., Tanriöver, Ö.Ö, Using a Subtractive Center Behavioral Model to Detect Malware, Secur. Commun. Networks, 7501894, 1-17, 2020.
[22] Mazini M., Shirazi B., Mahdavi I., Anomaly network-based intrusion detection system using a reliable hybrid artificial bee colony and AdaBoost algorithms, Journal of King Saud University - Computer and Information Sciences, 32 (10), 1206-1207, 2019.
[23] Balakrishnan S.M., Venkatalakshmi K., Kannan A., Intrusion Detection System Using Feature Selection and Classification Technique, IJCSA, 3 (4), 145, 2014.
[24] Torabi M., Udzir N.I., Abdullah M.T., Yaakob R.A., Review on Feature Selection and Ensemble Techniques for Intrusion Detection System, IJACSA, 12 (5), 538-553, 2021.
[25] Özkan Okay M., Aslan Ö., Eryiğit R., Samet R., SABADT: Hybrid Intrusion Detection Approach for Cyber Attacks Identification in WLAN, IEEE Access, 9, 157639-157653, 2021.
[26] Ambusaidi M.A., He X., Tan Z., Nanda P., Lu L.F., Nagar U.T., A Novel Feature Selection Approach for Intrusion Detection Data Classification, 2014 IEEE 13th International Conference on Trust, Security and Privacy in Computing and Communications, BeijingChina, 82-89, 24-26 Eylül, 2014.
[27] Chen C.W., Tsai Y.H., Chang F.R., Lin W.C., Ensemble feature selection in medical datasets: Combining filter, wrapper, and embedded feature selection results, Expert Systems, 37 (5), e12553, 2020.

[28] Song J., Feature selection for intrusion detection system, Ph.D. Thesis, Aberystwyth University, Department of Computer Science Institute of Mathematics, Physics and Computer Science, Penglais-UK, 2016.
[29] Kanimozhi V., Jacob P., Artificial Intelligence based Network Intrusion Detection with hyper-parameter optimization tuning on the realistic cyber dataset CSE-CIC-IDS2018 using cloud computing, ICT Express, 5 (3), 211-214, 2019.
[30] Latah M., Toker L., Towards an efficient anomaly-based intrusion detection for software-defined networks, IET Netw., 7, 453-459, 2018.
[31] Uğurlu M., Doğru İ. A., Arslan R.S., Detection and classification of darknet traffic using machine learning methods, Journal of the Faculty of Engineering and Architecture of Gazi University, 38 (3), 1737-1746, 2023.
[32] Aleesa, A., Younis, M. O. H. A. M. M. E. D., Mohammed, A. A., Sahar, N., Deep-intrusion detection system with enhanced UNSW-NB15 dataset based on deep learning techniques, Journal of Engineering Science and Technology, 16(1), 711-727, 2021.
[33] Choudhary, S., Kesswani, N., Analysis of KDD-Cup'99, NSL-KDD and UNSW-NB15 datasets using deep learning in IoT, Procedia Computer Science, 167, 1561-1573, 2020.
[34] Yousefi-Azar, M., Varadharajan, V., Hamey, L., Tupakula, U., Autoencoder-based feature learning for cyber security applications. In 2017 International joint conference on neural networks (IJCNN) (pp. 3854-3861), IEEE, 2017.
[35] Basati, A., Faghih, M., M., APAE: an IoT intrusion detection system using asymmetric parallel auto-encoder. Neural Computing and Applications, 35(7), 4813-4833, 2023.
[36] Altunay, H., C., Albayrak, Z., A hybrid CNN+ LSTM-based intrusion detection system for industrial IoT networks, Engineering Science and Technology, an International Journal, 38, 101322, 2023.
[37] Abedi, A., Khan, S. S., Fedsl: Federated split learning on distributed sequential data in recurrent neural networks, Multimedia Tools and Applications, 1-212, 2023.
[38] Bischl, B., Binder, M., Lang, M., Pielok, T., Richter, J., Coors, S., Lindauer, M., Hyperparameter optimization: Foundations, algorithms, best practices, and open challenges, Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 13(2), e1484, 2023.