Machine Learning system for detecting malicious traffic generated by IoT devices

Yurii Klots[1,*,†], Nataliia Petliak[1,†], Serhii Martsenko[2,†], Vitaliy Tymoshchuk[2,†] and Ievgen Bondarenko[3,†]

[1] *Khmelnytskyi National University, Cyber Security Department, 11, Instytuts'ka str., Khmelnytskyi, Ukraine*
[2] *Ternopil Ivan Puluj National Technical University, Ruska str., 56, Ternopil, Ukraine*
[3] *West Ukrainian National University, Lvivska str., 11, Ternopil, Ukraine*

Abstract

In this work, various combinations of artificial neural networks (CNN, LSTM, CNN-LSTM) are investigated for the analysis of outgoing traffic from IoT devices for the purpose of traffic classification and real-time attack detection. The focus is on the effectiveness of various combined approaches to data processing and analysis in IoT networks. The work uses KDDCup99, NSL-KDD, UNSW-NB15, WSN-DS and CICIoT2023 datasets for training and testing networks. To assess the reliability of the work of various algorithms, calculations of accuracy, specificity, sensitivity and other metric indicators determining the effectiveness of the proposed solutions were carried out.

Keywords

IoT, CNN, LSTM, CNN-LSTM, Outgoing traffic, Malicious traffic.

1. Introduction

The Internet of Things (IoT) has a wide variety of applications, which makes it unique among other types of computer networks. IoT networks can be built from devices of different types, characterized by different hardware, functionality and topology. Communication protocols can also vary from one implementation to another. Widespread use of IoT includes smart homes, intelligent transportation, and other areas of modern life. However, the incompatibility of security measures can create vulnerabilities that require special solutions to protect IoT networks from attacks. Intrusion detection can be an effective defense, but needs continuous improvement to ensure reliability. Innovations in IoT technologies are driving data management strategies, but also increasing the need for

CITI'2024: 2nd International Workshop on Computer Information Technologies in Industry 4.0, June 12–14, 2024, Ternopil, Ukraine
* Corresponding author.
† These authors contributed equally.
klots@khmnu.edu.ua (Y.Klots); npetlyak@khmnu.edu.ua (N.Petliak); marcenko@cei.net.ua (S.Martsenko); Tymoshchuk@tntu.edu.ua (V.Tymoshchuk); ye.bondarenko@wunu.edu.ua (I.Bondarenko)
0000-0002-5385-5761 (Y.Klots); 0000-0001-5971-4428 (N.Petliak); 0000-0003-3301-0216 (S.Martsenko); 0009-0007-2858-9434 (V. Tymoshchuk); 0000-0001-6856-4855 (I.Bondarenko)

security. One of the key challenges is the heterogeneity of the IoT network, which makes it difficult to deploy comprehensive security systems. Topology, communication protocols, and hardware can vary even within the same network, which increases the attack surface. The ever-changing nature of IoT networks requires the creation of intrusion detection systems that are effective in real-time and robust to changes in the network.

In [1], an intrusion detection model is considered, which combines the advantages of spiking neural network (SNN) and convolutional neural networks (CNN) with the help of rational algorithm design. This model allows efficient use of resources, which ensures adaptability to limited computing capabilities.

In [2], the authors propose the use of a multi-scale convolutional feature fusion network augmented with a Convolutional Block Attention Module (MCF-CBAM) for IoT traffic classification. Their approach includes the following features: parallel convolution obtains spatial characteristics from traffic data; the attention module mutes less informative features while boosting the most discriminative ones to provide focused learning on key features.

The authors of [3] propose a sequential approach to feature selection using an optimized extreme learning machine (ELM) with a support vector machine (SVM) classifier, where a genetic algorithm (GA) is used to optimize the ELM weights. The optimized data set is used to classify traffic for intrusion detection in an IoT environment.

In [4], the authors demonstrate the synthesis of Decisive Red Fox (DRF) optimization with a machine learning algorithm. Based on the optimized characteristics, the DBRF classification process is used to identify and classify intrusion types.

The authors of [5] propose an intrusion detection system and configuration of dynamic rules SecureFlow for IoT environments. This implementation is based on knowledge and data, forming a two-level system. An environment with Software-defined Networking (SDN) support allows you to configure rules according to detected incidents.

In [6], a hybrid deep learning model is proposed for detecting botnet attacks in IoT networks. The two-stage hybrid model analyzes the network traffic data obtained from three parallel sensors and detects the simultaneous characteristics of the attack traffic. Features are extracted using the long-term memory-based autoencoder (LSTM-AE) using the NCC-2 Simultaneous Botnet Dataset. LSTM-AE is trained on data from multiple sensors to model temporal characteristics. The type of attack is identified using multi-class classification using an ensemble learning algorithm with extreme gradient boosting (XGBoost).

G. Parimala and R. Kayalvizhi [7] proposed a hybrid deep learning model (HDLM) based on IoT device intrusion detection and prevention, where important features are taken from the KDDCup99 and NSL-KDD datasets using a forward feature selection algorithm (FFSA). . The features are then fed into the HDLM classifier. The proposed HDLM is a combination of Ellman Recurrent Neural Network (ERNN) and Subtraction Based Optimizer (SABO).

The authors of [8] analyzed three different models for intrusion detection in the Industrial Internet of Things (IIoT) network using deep learning architectures: CNN, long-short-term memory (LSTM), and a combination of CNN-LSTM, which were created based on their hybrid combination. According to the obtained results, the CNN-LSTM model

demonstrated higher accuracy for the binary and multi-class classification processes in the UNSW-NB15 and X-IIoTID datasets compared to the other two models used in this study, namely CNN and LSTM.

[9] presents an IDS architecture based on CNN and LSTM algorithms. The research result of CNN-LSTM compared to CNN and machine learning models for both balanced and unbalanced data showed better performance in detecting IoT security attacks using the UNSW-NB15 dataset.

Shreeya Jain et al. [10] demonstrate a hybrid IoT intrusion detection model by combining Deep Learning (DL), CNN, and LSTM techniques to achieve better attack detection accuracy. The model is trained and evaluated using two different datasets, namely UNSW-NB15 1 and NSL-Botnet 2.

[11] proposed a DL model for detecting anomalies in IoT networks using a recurrent neural network (RNN). LSTM, Bidirectional LSTM and Gated Recurrent Unit (GRU) methods are used to implement the proposed model. A hybrid DL model using CNN and RNN networks was proposed. A DL model for binary classification using LSTM, BiLSTM and GRU based approaches was also proposed. The described deep learning models are tested using NSLKDD, BoT-IoT, IoT-NI, IoT-23, MQTT, MQTTset and IoT-DS2 datasets.

[12] presents a hybrid intrusion detection model (HIDM) that uses Optimized CNN-LSTM (OCNN-LSTM) and Transfer learning (TL) for IIoT networks. The proposed model uses an optimized CNN using advanced CNN parameters using the Gray wolf optimizer (GWO) method, which tunes the CNN parameters and helps to improve the prediction accuracy of the model. The transfer learning model helps train the model and transfers the knowledge to the OCNN-LSTM model. The TL method improves the learning process by obtaining the necessary knowledge from the OCNN-LSTM model. Classification analysis was performed on several classes of different datasets (ToN-IoT and UNW-NB15).

[13] proposes an intrusion detection system (IDS), namely SafetyMed, which combines CNN and LSTM to defend against intrusion from sequential and grid data. SafetyMed is an IDS that protects Internet of Medical Things devices from malicious data and persistent network traffic.

In [14], the DL model for detecting intrusions into the IoT network is described. To obtain the sequence properties of the data stream through CNN, it combines a control mechanism with an LSTM network. The paper used a feature selection strategy to train the classifiers on the most significant correlation features while avoiding lost results during training to obtain the best results. The proposed strategy focuses on binary classification using DL methods.

In the considered works, machine learning methods are used, which mostly give a good result, but they are aimed at the analysis of incoming traffic to the network (Fig.1). When changing the type of attack, the class of attacked devices, the level of detection of attacks decreases significantly [15,16].

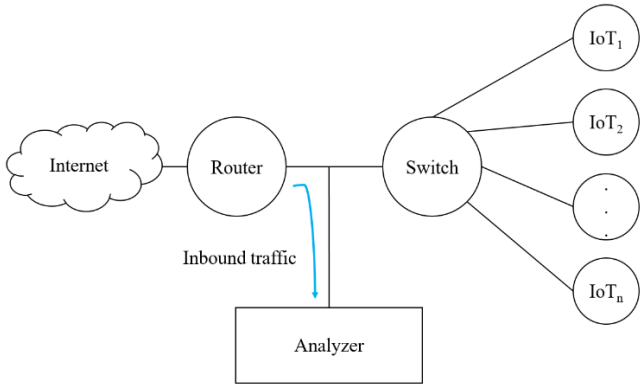

Figure 1: Classic traffic analyzer

One of the reasons for attacks on IoT systems is to create a network of bots or third-party controlled devices to carry out large-scale attacks on government and commercial systems.

In order to prevent the spread of an attack from the network, we will represent the internal network as a black box and analyze the outgoing traffic in order to detect attacks from the system (Fig.2).

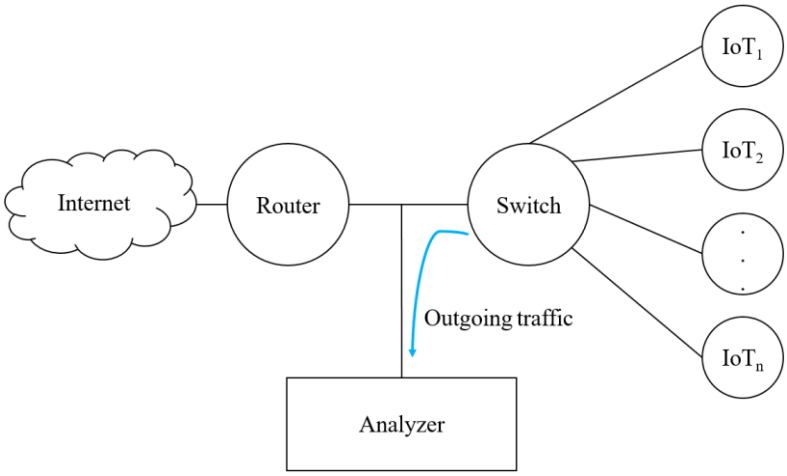

Figure 2: Proposed traffic analyzer

The analysis carried out in [1-14] shows that CNN, LSTM and a combination of the specified neural networks show the best result for investigating traffic and detecting malicious actions related to IoT. Works [15,16] show the expediency of analyzing the outgoing traffic.

Therefore, it is advisable to conduct a study on the use of CNN, LSTM and their combinations on different data sets to detect malicious actions from IoT devices.

2. Data sets for training neural networks

The standard datasets KDDCup99, NSL-KDD, UNSW-NB15, WSN-DS and CICIoT2023 were used in this study. These sets make it possible to evaluate the effectiveness of the developed model for detecting malicious traffic in the network.

The KDDCup99 dataset contains recordings from real network traffic, including normal traffic and various types of attacks. It is one of the most widely used datasets for evaluating anomaly detection techniques. Since 1999, KDDCup99 has been the most widely used dataset for evaluating anomaly detection methods. Based on data collected by the DARPA program, which is based on approximately 4 gigabytes of tcpdump data from seven weeks of network traffic and approximately 5 million connections. The test data for a two-week period is about 2 million connection records. The dataset consists of 4,94,021 data points and 42 features labeled as normal or attacks, with only one specific attack type. It is categorized as a type of attack. Attacks are classified into one of the following four categories: Denial of Service(DoS)attacks, User-to-Roo (U2R), Remote tolocal(R2L) attacks, Probingattacks.

The NSL-KDD dataset is an improved version of the original KDDCup99 dataset. It was designed to address some of the limitations and shortcomings of the KDDCup99 dataset in the field of intrusion detection. The dataset was specifically designed to evaluate intrusion detection systems, particularly in the context of network security.

The UNSW-NB15 dataset consists of raw network packets. The dataset contains nine types of attacks, including phaser, analysis, backdoor, DoS, exploit, general purpose, reconnaissance, shellcode, and worm. The dataset consists of 2,540,044 records stored in four CSV files, and the training set and test set contain 175,341 and 82,332 records, respectively. This dataset is used for a variety of research activities related to intrusion detection, network forensics, privacy protection, and threat analysis in various systems such as networked systems, Internet of Things (IoT), SCADA, Industrial IoT, and Industry 4.0.

WSN-DS is a data set specially created for detecting attacks in wireless sensor networks (Wireless Sensor Networks, WSN). The ns-2 simulation environment was used for data collection. The dataset includes 23 features obtained using the LEACH routing protocol that describe the state of each sensor node in the wireless network. The WSN-DS dataset consists of 374,661 tests divided into four attack types. The tests are divided into five different classes: Blackhole, Grayhole, Flooding, TDMA and Typical, with four of them dealing with different types of DoS attacks. The dataset tests are divided into five different classes, four of which are related to different types of DoS attacks.

The CIC IoT 2023 dataset is a real-world testbed for large-scale Internet of Things (IoT) attacks. Its primary goal is to provide an expanded and novel IoT attack dataset to support the development of security analytics applications in real-world IoT environments. To achieve this goal, 33 attacks were performed on an IoT topology consisting of 105 devices. These attacks were divided into seven categories, including DDoS, DoS, Recon, Web Attacks, Brute Force, Spoofing and Mirai. All attacks were performed by malicious IoT devices that target other IoT devices.

Preparing datasets for ML involves several important steps to ensure that the data is appropriate for effectively training a model to detect malicious network traffic (Fig.3). In

the first step, a raw data set was loaded into the system. The data set then underwent a coding step, which was necessary to convert the categorical variables into a format understandable by the model. The data were then normalized to ensure that the dimensionality of the input data did not negatively affect the learning process. The next step was to select features. At the end, the dataset is split into training and testing sets.

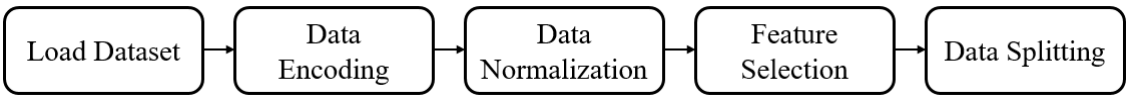

Figure 3: Preparing datasets

The KDDCup99 dataset includes 5209460 records. For training neural networks, 80% of the records from the total data set, namely 4167568 records, were selected. There are 20% of records left for testing, namely 1041892 records.

The NSL-KDD dataset consists of 5209458 records. 4,898,431 records are used for training, of which 3,925,650 records are marked as malicious and 972,781 records are marked as normal, reflecting real-world scenarios where malicious traffic often exceeds normal traffic. The test set consisted of 311027 records, where 250436 records represent attacks and 60591 records represent normal interactions, creating a realistic challenge for ML.

The UNSW-NB15 data set is smaller compared to previous ones, consisting of 257,673 records. 175341 records from the dataset were used for training. The test set contained 82332 records in total, where the majority of interactions, namely 78832 records, are malicious and 3500 records of normal traffic. It should be noted that the data sets are not balanced in terms of the number of records of normal and malicious traffic, so the accuracy parameter estimate is not informative.

The WSN-DS dataset, which is designed for wireless sensor networks, contains 374,661 records. They were divided by 60% for training, resulting in 224,796 records, of which 204,039 were identified as normal traffic. For testing, 40% was used, namely 149865 records in total, of which 136027 were identified as normal traffic. It should be noted that the data sets are not balanced in terms of the number of records of normal and malicious traffic, so the accuracy parameter estimate is not informative.

The CICIoT2023 dataset focuses on malicious activities and contains a total of 45588384 malicious entries, while 1098195 entries are identified as normal traffic. 36470707 malicious records and 878556 normal traffic records were selected for training, which is 80% of the total number of malicious records. 9117677 malicious records and 219639 records of normal traffic were used for testing.

3. Neural networks for analyzing outgoing traffic from IoT

The CNN network is effective in analyzing network traffic because it excels at automatically detecting and learning complex data patterns. The working principle of CNN for network traffic analysis:

1. Removal of functions.
2. Activation functions.

After each convolution operation, an activation function is applied to introduce nonlinearity. The ReLU (Rectified Linear Unit) activation function for ML was used due to its efficiency and computational simplicity.

$$f(x) = \max(0, x) \quad (1)$$

ReLU is fast because it replaces all negative values with zero, thereby simply "turning off" some neurons, which helps create sparse networks and potentially speeds up computation.

3. Combining layers.
4. Fully connected layers.
5. Initial level.

The output layer uses a softmax activation function to classify incoming network traffic into categories such as normal and malicious.

$$softmax(x_i) = \frac{e^{x_i}}{\sum_j e^{x_j}} \quad (2)$$

In general, the algorithm for preparing the CNN-LSTM neural network is shown in the figure 4. The main steps are: data preparation and processing, training of the CNN-LSTM network model, and evaluation of the test results using the confusion matrix.

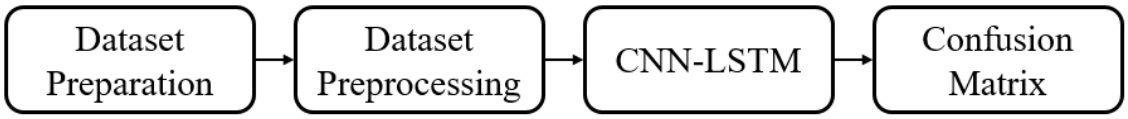

Figure 4: Algorithm for preparing the CNN-LSTM neural network

The figure 5 illustrates a neural network architecture that combines CNN and LSTM. Input values that have been preprocessed are received at the Input input. C1, C3 are convolutional layers that are responsible for feature extraction, highlight important characteristics in the data. S2 and S4 are pooling levels, specifically maximum size pooling levels that follow some convolution layers. Pooling layers reduce the spatial dimensions of the input volume for the next convolutional layer, which reduces the number of parameters and computations in the network, thereby controlling reconfiguration. GlobalMaxPool is a global maximum pool that further reduces each feature map to a single number by taking the maximum value of the feature map sizes while keeping the most significant feature response. This helps reduce the dimensionality of the data before passing it to the LSTM layer, allowing the network to efficiently process data sequences. Next, the data is passed to the LSTM layer. L6 is a fully connected layer, which means that every neuron in this layer is connected to all neurons in the previous layer. This layer combines the features obtained by CNN and LSTM to make a decision. Softmax output is the last output level with an activation function.

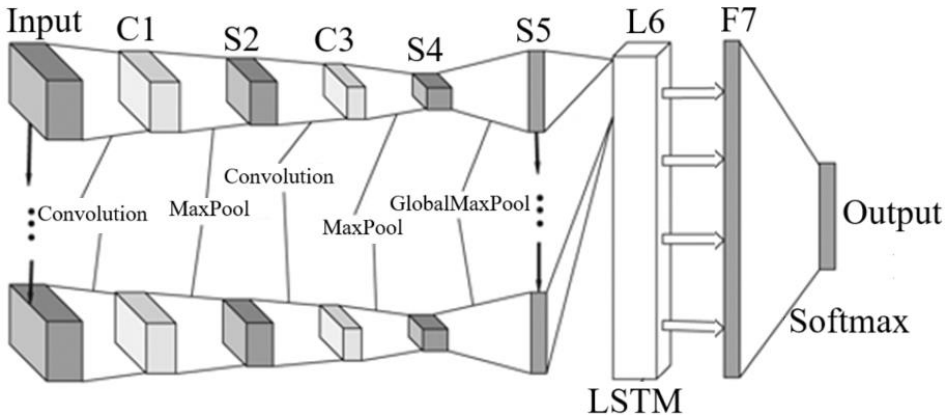

Figure 5: A neural network architecture that combines CNN and LSTM

4. Evaluation of the reliability of the use of neural networks

To assess the reliability of the developed system, a confusion matrix was used (Fig.6). True Positive (TP) indicates the number of correctly identified malicious network traffic flows. True Negative (TN) indicates the number of correctly identified normal network traffic flows. False Positive (FP) is the number of times the system detects malicious traffic, even though the traffic is normal. False Negative (FN) the number of system triggers where the traffic flow was classified as normal even though it was malicious. The indicated results allow the calculation of the following performance evaluation indicators: accuracy, precision, recall, specificity and F-score.

		Predicted Class	
		Positive	**Negative**
Actual class	**Positive**	**True Positive**	**False Negative (Type II Error)**
	Negative	**False Positive (Type I Error)**	**True Negative**

Figure 6: Confusion matrix

Accuracy allows you to calculate the ratio of the total number of valid hits for the entire data set:

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \tag{3}$$

Precision measures how accurately the system classifies objects or events as malicious when it detects them as such. This metric is calculated as the ratio of correctly identified

malicious objects or events to all objects or events that the system identified as malicious:

$$Precision = \frac{TP}{TP + FP} \quad (4)$$

Recall determines the system's ability to detect all existing malicious sessions without missing any of them. It indicates how effectively the system responds to real threats:

$$Recall = \frac{TP}{TP + FN} \quad (5)$$

Specificity is a metric that measures the effectiveness of a system in correctly identifying benign objects or events. It is defined as the ratio of the number of correctly identified non-malicious objects or events to the total number of non-malicious objects or events:

$$Specificity = \frac{FP + FN}{TP + FP + TN + FN} \quad (6)$$

The F-score represents a weighted average of the true positive result and accuracy, where:

$$F - score = \frac{2 \times Recall \times Precision}{Recall + Precision} \quad (7)$$

The results of testing CNN, LSTM and CNN-LSTM networks with different data sets are shown in table (1-3).

Table 1
Quality metrics for the CNN network

CNN	TP	TN	FP	FN
KDDCup99	654812	258258	60769	68053
NSL-KDD	235840	43835	15279	16073
UNSW-NB15	3297	78035	497	503
WSN-DS	12370	129344	3947	4194
CICIoT2023	8251354	158392	439470	488087

Table 2
Quality metrics for LSTM networks

LSTM	TP	TN	FP	FN
KDDCup99	674142	264400	45675	57675
NSL-KDD	239411	42935	13364	15317
UNSW-NB15	3341	78192	364	435
WSN-DS	12425	131976	2344	3120
CICIoT2023	8292540	169734	457622	417420

Table 3
Quality metrics for CNN-LSTM networks

CNN-LSTM	TP	TN	FP	FN
KDDCup99	718514	234268	42756	46354
NSL-KDD	244493	38721	13874	13939
UNSW-NB15	3358	78542	197	235
WSN-DS	13572	134280	935	1078
CICIoT2023	8741953	203457	154384	237507

Performance indicators for CNN, LSTM, and CNN-LSTM networks when training and testing using KDDCup99, NSL-KDD, UNSW-NB15, WSN-DS, and CICIoT2023 datasets are shown in Table 4.

Table 4
Performance indicators

Data set	Network type	Acccuracy	Precision	Recall	Specificity	F-score
KDDCup99	CNN	0,88	0,92	0,91	0,12	0,91
	LSTM	0,9	0,94	0,92	0,10	0,93
	CNN-LSTM	0,91	0,94	0,94	0,09	0,94
NSL-KDD	CNN	0,90	0,94	0,94	0,10	0,94
	LSTM	0,91	0,95	0,94	0,09	0,94
	CNN-LSTM	0,91	0,95	0,95	0,09	0,95
UNSW-NB15	CNN	0,99	0,87	0,87	0,01	0,87
	LSTM	0,99	0,90	0,88	0,01	0,89
	CNN-LSTM	0,99	0,94	0,93	0,01	0,94
WSN-DS	CNN	0,95	0,76	0,75	0,05	0,75
	LSTM	0,96	0,84	0,80	0,04	0,82
	CNN-LSTM	0,99	0,94	0,93	0,01	0,93
CICIoT2023	CNN	0,90	0,95	0,94	0,10	0,95
	LSTM	0,91	0,95	0,95	0,09	0,95
	CNN-LSTM	0,96	0,98	0,97	0,04	0,98

The evaluation of the effectiveness of the test results is demonstrated in the form of charts with a division by data sets.

The CNN-LSTM network on the KDDCup99 data set (Fig.7) demonstrated: accuracy, recall and F-score 3% higher than the CNN network; accuracy and F-score by 1% more than the LSTM network.

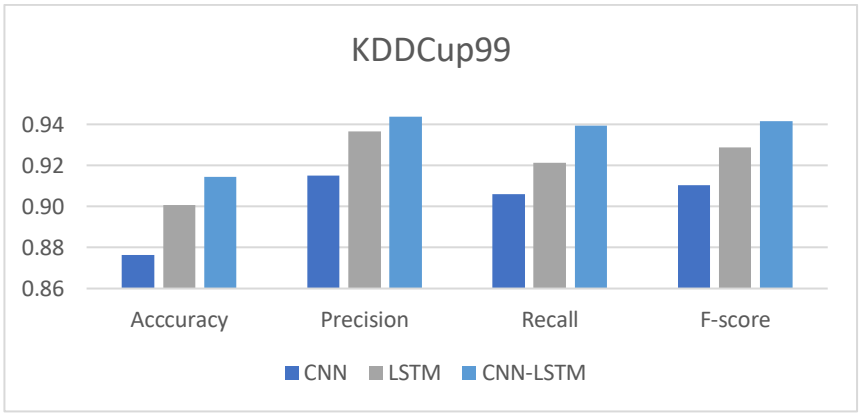

Figure 7: Performance evaluation for the KDDCup99 dataset

The CNN-LSTM network on the NSL-KDD data set (Fig.8) demonstrated: accuracy, precision, recall and F-score 1% more than the CNN network; recall and F-score is 1% higher than that of the LSTM network.

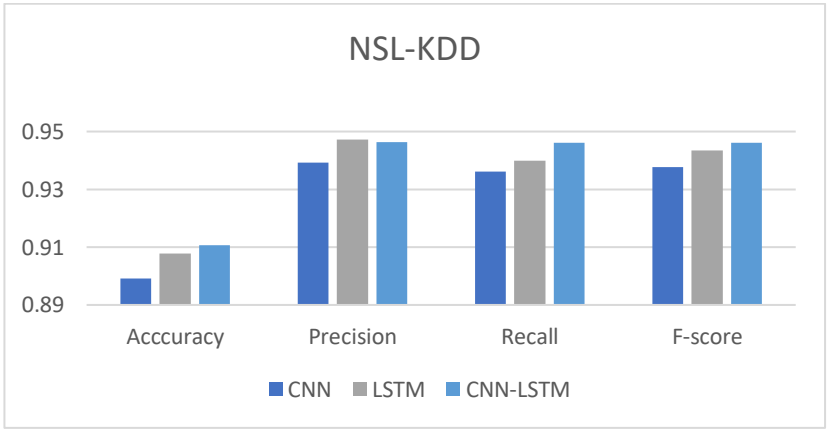

Figure 8: Performance evaluation for the NSL-KDD dataset

The CNN-LSTM network on the UNSW-NB15 data set (Fig.9) demonstrated: precision by 7%, recall by 6% and F-score by 7% more than the CNN network; precision by 4%, recall by 5% and F-score by 5% more than the LSTM network.

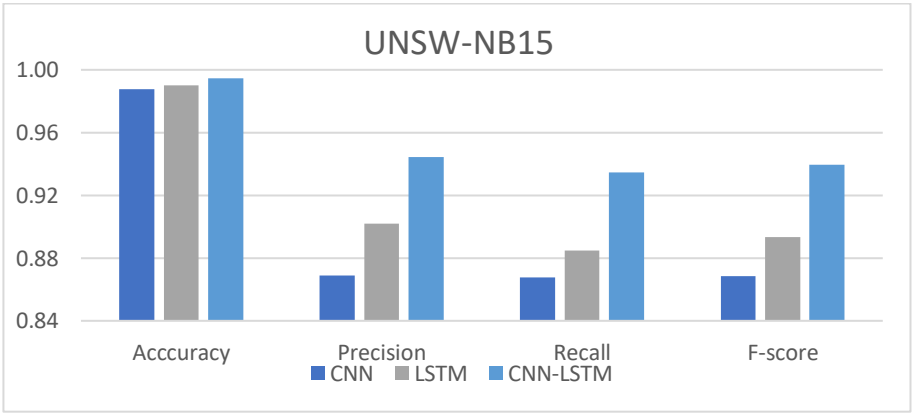

Figure 9: Performance score for the UNSW-NB15 data set

The CNN-LSTM network on the UNSW-NB15 data set (Fig.10) demonstrated: accuracy by 4%, precision by 18%, recall by 18% and F-score by 18% more than the CNN network; accuracy by 3%, precision by 10%, recall by 13% and F-score by 11% more than in the LSTM network.

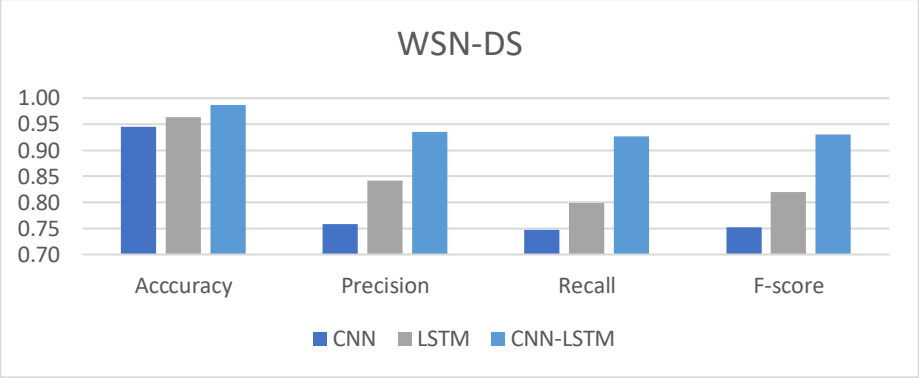

Figure 10: Performance evaluation for the WSN-DS dataset

The CNN-LSTM network on the CICIoT2023 data set (Fig.11) demonstrated: accuracy by 6%, precision by 3%, recall by 3% and F-score by 3% more than the CNN network; accuracy by 5%, precision by 3%, recall by 2% and F-score by 3% more than in the LSTM network.

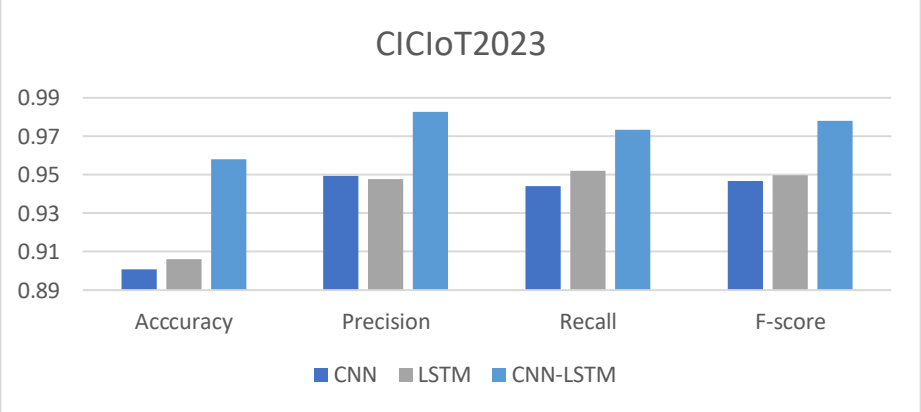

Figure 11: Performance evaluation for the CICIoT2023 dataset

Specificity in the CNN-LSTM network when tested on the KDDCup99 dataset showed a 3% better result compared to CNN and a 1% better result compared to LSTM. Specificity in the CNN-LSTM network when tested on the NSL-KDD dataset showed a 1% better result compared to CNN. The specificity of the CNN-LSTM network when tested on the UNSW-NB15 dataset showed the same result compared to CNN and LSTM. The specificity of the CNN-LSTM network when tested on the WSN-DS dataset showed a 4% better result compared to CNN and a 3% better result compared to LSTM. Specificity in the CNN-LSTM network when tested on the CICIoT2023 dataset showed a 6% better result compared to CNN and a 5% better result compared to LSTM.

The figure 12 shows the specificity parameter for all datasets and networks.

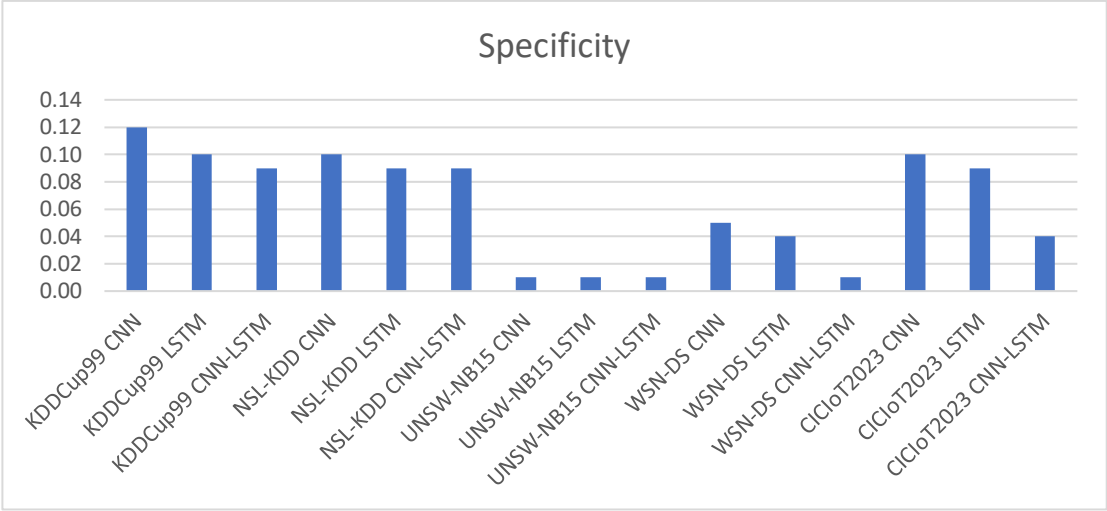

Figure 12: The specificity parameter for all datasets and networks

Conclusion

In view of the results of the conducted research, taking into account the types of attacks, the traffic from the implementation of which is present in the analyzed data sets, it can be concluded that the CNN-LSTM combination gives the highest reliability results and the lowest error results. Therefore, it is advisable to use CNN-LSTM and train it on the analyzed data sets for the detection system of the original malicious traffic.

References

[1] Wang, Z., Ghaleb, F.A., Zainal, A. et al. An efficient intrusion detection model based on convolutional spiking neural network. Sci Rep 14, 7054 (2024). https://doi.org/10.1038/s41598-024-57691-x

[2] Liao, N., Guan, J. Multi-scale Convolutional Feature Fusion Network Based on Attention Mechanism for IoT Traffic Classification. Int J Comput Intell Syst 17, 36 (2024). https://doi.org/10.1007/s44196-024-00421-y

[3] Maseno, E.M., Wang, Z. Hybrid wrapper feature selection method based on genetic algorithm and extreme learning machine for intrusion detection. J Big Data 11, 24 (2024). https://doi.org/10.1186/s40537-024-00887-9

[4] Rabie, O.B.J., Selvarajan, S., Hasanin, T. et al. A novel IoT intrusion detection framework using Decisive Red Fox optimization and descriptive back propagated radial basis function models. Sci Rep 14, 386 (2024). https://doi.org/10.1038/s41598-024-51154-z

[5] Amritpal Singh, Pushpinder Kaur Chouhan and Gagangeet Singh Aujla. SecureFlow: Knowledge and data-driven ensemble for intrusion detection and dynamic rule configuration in software-defined IoT environment. Ad Hoc Networks, 156 (2024). https://doi.org/10.1016/j.adhoc.2024.103404

[6] Belkacem, S. (2024). Simultaneous botnet attack detection using long short term memory-based autoencoder and XGBoost classifier. International Journal of Safety

and Security Engineering, Vol. 14, No. 1, pp. 155-163. https://doi.org/10.18280/ijsse.140115

[7] G. Parimala and R. Kayalvizhi. Improved Elman Deep Learning Model for Intrusion Detection System in Internet of Things. Journal of Internet Services and Information Security, 14 (2024). https://doi.org/10.58346/JISIS.2024.I1.008

[8] Hakan Can Altunay, Zafer Albayrak. A hybrid CNN+LSTM-based intrusion detection system for industrial IoT networks. Engineering Science and Technology, an International Journal, vol 38 (2023). https://doi.org/10.1016/j.jestch.2022.101322

[9] Rasha Almarshdi, Laila Nassef, Etimad Fadel, Nahed Alowidi. Hybrid Deep Learning Based Attack Detection for Imbalanced Data Classification. Intelligent Automation \& Soft Computing, vol 35 (2023). https://doi.org/10.32604/iasc.2023.026799

[10] Shreeya Jain, Pranav M. Pawar, Raja Muthalagu. Hybrid intelligent intrusion detection system for internet of things. Telematics and Informatics Reports, Volume 8, 2022. https://doi.org/10.1016/j.teler.2022.100030.

[11] I. Ullah and Q. H. Mahmoud, "Design and Development of RNN Anomaly Detection Model for IoT Networks," in IEEE Access, vol. 10, pp. 62722-62750, 2022, doi: 10.1109/ACCESS.2022.3176317

[12] Lilhore UK, Manoharan P, Simaiya S, Alroobaea R, Alsafyani M, Baqasah AM, Dalal S, Sharma A, Raahemifar K. HIDM: Hybrid Intrusion Detection Model for Industry 4.0 Networks Using an Optimized CNN-LSTM with Transfer Learning. Sensors. 2023; 23(18):7856. https://doi.org/10.3390/s23187856

[13] Faruqui N, Yousuf MA, Whaiduzzaman M, Azad A, Alyami SA, Liò P, Kabir MA, Moni MA. SafetyMed: A Novel IoMT Intrusion Detection System Using CNN-LSTM Hybridization. Electronics. 2023; 12(17):3541. https://doi.org/10.3390/electronics12173541

[14] Zakariah M, AlQahtani SA, Al-Rakhami MS. Machine Learning-Based Adaptive Synthetic Sampling Technique for Intrusion Detection. Applied Sciences. 2023; 13(11):6504. https://doi.org/10.3390/app13116504

[15] Klots, Y., Titova, V., Petliak, N., Cheshun, V., Salem, A.-B.M. Research of the Neural Network Module for Detecting Anomalies in Network Traffic. CEUR Workshop Proceedings, 2022, 3156, pp. 378-389.

[16] Y. Klots, N. Petliak and V. Titova. Evaluation of the efficiency of the system for detecting malicious outgoing traffic in public networks. 13th International Conference on Dependable Systems, Services and Technologies (DESSERT), Athens, Greece, 2023, pp. 1-5, https://doi.org/10.1109/DESSERT61349.2023.10416502.