





Using Hybrid Deep Learning Approach to Enhanced Network Intrusion Detection with Spatial-Temporal Feature Integration

Jane J. Stephan¹, Mohammed Q. Mohammed^{2,3*}

¹ Department of Information Technology, Al-Esraa University, Baghdad 10001, Iraq

² Department of Cybersecurity Engineering, Al-Esraa University, Baghdad 10001, Iraq

³ Department of Informatics Systems Management, University of Information Technology and Communications, Baghdad 10001, Iraq

Corresponding Author Email: dr.mohammed@esraa.edu.iq

Copyright: ©2024 The authors. This article is published by IETA and is licensed under the CC BY 4.0 license (<http://creativecommons.org/licenses/by/4.0/>).

<https://doi.org/10.18280/isi.290435>

ABSTRACT

Received: 18 February 2024

Revised: 23 July 2024

Accepted: 31 July 2024

Available online: 21 August 2024

Keywords:

Intrusion Detection Systems (IDS), Difficult-Set-Sampling-Technique (DSST), Deep-Convolutional-Generative-Adversarial-Networks (DCGAN), EESNN, Telecommunications Network Internet of Things (ToN-IoT), CICIDS2019, SAT-Net

Intrusion Detection Systems (IDS) play a vital role in network security by detecting and preventing malicious activities. The network intrusion data is integrated into a vast number of common occurrences due to the dynamic and ever-changing networking environment. This results in a scarcity of training cases for models and detection outcomes, accompanied by a significant percentage of false detections. Our suggested Network-IDS addresses the issue of data imbalance by integrating Deep Learning Networks (DLN) via hybrid sampling. We begin by collecting out-of-the-ordinary samples from the majority and eliminating them using the Difficult-Set-Sampling-Technique method, which stands for Difficult-Set-Sampling-Technique (DSST). Next step is to increase the minority group's sample size using (DCGAN) means Deep-Convolutional-Generative-Adversarial-Networks. Step two involves building a model for a deep neural network to extract geographical features using DenseNet169, in addition, we utilize SAT-Net to capture features of temporal. This approach effectively represents the unique attributes of the dataset. Lastly, we deployed the EESNN to identify assault types. In addition to that, we conducted tests on the latest and most extensive intrusion datasets, the Telecommunications Network Internet of Things (ToN-IoT) dataset as well as the CICIDS2019 dataset, to verify of proposed approach. The outcome demonstrates that our recommended structure surpasses similar efforts in terms of accuracy, false alarm rate, recall, and precision. The findings indicate that our proposed system is superior to other attempts of a similar kind in terms of accuracy, false alarm rate, recall, and precision. We will provide a detailed explanation of this in the comparative section.

1. INTRODUCTION

The Internet of Things (IoT) facilitates communication between devices by utilizing various communication protocols, allowing them to exchange data. Smart systems consist of several components, including servers, software, control panels, sensors, and network infrastructure [1]. The Internet of Things technology transforms passive objects into smart ones by establishing a digital interconnection layer to address common real-world problems. Various ubiquitous global systems are now using IoT, including smart homes and workplaces [2], online education, economics, marketing, and smart cities [3, 4]. By 2035, there will be a total of one trillion Internet of Things (IoT) devices connected to the internet, and an additional 75 billion devices are projected to be online by 2025 [5].

The widespread use of Internet-connected devices ensures that the network will inevitably see a large increase in data, namely in the form of traffic containing substantial amounts of information [6]. Substantial and confidential information has

been divulged. Empirical evidence consistently indicates that cybercriminals will develop a keen interest in sensitive information. Those who possess the shared information are subject to substantial liability in the event of an unauthorized intrusion, including larceny or manipulation. Integrity, privacy, confidentiality, and utility must be ensured. Consumers and manufacturers of Internet of Things devices must prioritize the elimination of security hazards. The challenge of designing a security solution for the network's connected devices, or the Internet of Things, is challenging due to the devices' unique characteristics and limited resources. Therefore, the 'hackers', or rather cybercriminals, are shifting their attacks more often to the IoT devices.

Al-Rubaye and Türkben [7] has identified that gadgets can be undermined through diverse approaches that include the development and modification of software and hardware, injection of improper code and setting up of viruses. By these actions, they in fact gain unauthorized access to the IoT network iOS stands for Internet of Things.

1.1 Network Intrusion Detection Systems (NIDS)

NIDS stands for Network Intrusion Detection Systems which are components of robust frames used in network security to detect intrusion and abnormalities in network traffic. These systems examine the incoming and the outgoing packets of a network and determine whether or not they match attack signatures already stored in the system's database, or apply the anomaly detection method that examines the irregularity within the system. The major goal of NIDS is to alert the system administrator on unauthorized activities as soon as possible to prevent network tampering and information menace [8].

NIDS operate in two primary modes: It is classified into two parts which include; signature-based detection and anomaly-based detection. In signature-based detection, traffic in a network is scanned using prescribed signatures of existing threats. This method is very efficient in identifying known attacks but bears low efficiency when it comes to unknown or emerging forms of attacks. Another category of detection techniques is known as anomaly-based, which creates a model of normal network traffic and seek to find any traffic that deviates from the model as anomalous traffic. As such, although this method excels in discovering new attacks, it yields a relatively larger number of false positives [9].

Recent research has revealed that deep learning-based NIDSs have better accuracy than the machine learning ones, but they fail to detect the assaults that have lower traffic because of the dataset biases [10]. Modern intrusion detection datasets often have a small number of attack instances compared to the instances of regular traffic; thus, datasets have an unbalanced class distribution. This hinders the detection of particular types of assaults and diminishes the effectiveness of NIDS [11]. Low rates of detection and high rates of false alarms indicate ineffectual performance. Recent NIDS investigations, however, have disregarded unequal data [12-17].

This study aims to address class imbalances in NIDS by implementing a hybrid sampling strategy, reducing majority samples utilizing the Difficult Set Sampling Technique as well as increasing minority samples utilizing a Deep Convolutional Generative Adversarial Networks model. When considering attributes of geography, the DenseNet 169 model is considered optimal, while SAT-Net is considered more suitable for temporal features. In terms of the classification of assaults, the EESNN operates with remarkable efficiency. When applied to the ToN-IoT as well as CICIDS 2019 attack detection datasets, the proposed method outperformed earlier approaches.

By enhancing the detection capabilities of NIDS through the integration of spatial-temporal features and addressing class research contributes to the development of more robust and effective intrusion detection mechanisms, ultimately enhancing the security of IoT networks.

2. LITERATURE REVIEW

2.1 Deep-learning-based IDS

Deep learning is a subdivision of machine learning that use multiple layers of representation to accurately depict intricate relationships and concepts. It plays a crucial role in protecting networks from cyberattacks, making it a vital element of cybersecurity. IDSs have made significant progress by

leveraging deep learning techniques, which have been made possible by breakthroughs in computer vision, image processing, and natural language processing. DL has been popular among researchers because of its very efficient hierarchical feature representations and ability to capture long-term temporal patterns. These hierarchical and heuristic search structures are highly effective.

As a result, DL approaches are being investigated as a means to improve IDS intelligence, even if there is a dearth of studies that compare them to publicly accessible datasets. High-quality learning for complicated data processing is made possible by DL's sophisticated structural design, and a solid system basis is provided by parallel processing technology [16].

The 2018 dataset from the CICIDS is extensively utilized since it fixes concerns with the 2017 dataset. It includes various types of traffic and real-world network traffic, making it popular. However, a significant issue needs to be addressed, as it results in a high-class imbalance that misleads the classifier [17].

2.2 Related work

ML and DL approaches have been widely used in network security due to their ability to distinguish data [18-22]. Other researchers have used various techniques for identification (IDS) using KNN and SVM on various datasets to evaluate the efficiency of these algorithms on the NSL-KDD dataset [23, 24].

According to research [25], an integrated ID system might be developed by combining the ANN with correlation-based feature selection. The authors performed an empirical investigation utilizing the UNSW-NB15 and NSL-KDD ID datasets. A proposed ID system was an RF-based system in the study by Siddiqui and Naahid [26], whereas an identification system utilizing numerous conventional ML classification algorithms was suggested by Binbusayyis and Vaiyapuri [27]. According to the again lower FP and DR of the ID system, it is apparent that earlier attempts of identification (ID) were not very effective for classification. The experiments were performed by Bhavani et al. [28] on the synthesized dataset KDD CUP'99 benchmark and the non-symmetric deep auto-encoder to counter problems related to network intrusion detection.

In another work, an architecture for Network Intrusion Detection Systems (NIDS) was proposed based on the deep learning technique that involved the use of One Dimensional Convolutional Neural Network (1D-CNN). The authors used CICIDS2017 dataset for their testing and for the recommendation, they developed a NIDCS using Ada Boost [29, 30]. The authors were able to identify sectarian irregularities in the network through the UNSW-NB 15. Based on the test results, it is evident that the proposed method can be applied in identifying various types of computer network violation. Domain learning refers to that branch of machine learning where hidden layers help in identifying the properties of the network in question. These approaches are considered better than ML because of their proper frameworks and the CAP ability of autonomous data comprehension and dataset generation [24]. Well, the kind of identification that is developing very fast nowadays is DL, or deep learning, because the studies revealed that it works better than other existing methods.

Another study [31] discovered that DL, which was utilized

for anomaly-based flow identification on DNN, could also be applied to the identification of network anomalies. Decentralized cloud-based intrusion detection was proposed by Kim et al. [21]. The Naive Bayes model identified outliers to start preprocessing. RF then identified each attack pattern in preparation for multi-classification. On the CICDDS-001 dataset, research was conducted regarding FPR and precision. An IDS that supports wireless meshes and incorporates multiple vector classifiers [32]. The authors employed support vector machine (SVM) classification and genetic algorithm-based feature selection to optimize efficiency. Both a standard intrusion dataset and an intrusion dataset derived by WMN were employed in the network simulator-3 (NS3) to analyze and simulate the system. For model evaluation, the CICIDS 2017 intrusion dataset was utilized. The ensemble-based intrusion detection system network anomaly identification method was proposed in reference [33]. This approach uses learning and predictions to classify anomalies. The ANOVA F-test was used in conjunction using univariate feature selection to examine feature performance and the correlations between class labels as well as data variables. the numbers [34, 35].

Researchers have developed various intrusion detection (IDS) feature selection methods, including automated machine learning models, Kalman filters, Bayesian optimizers, and auto-encoders. These methods have been tested on publicly accessible datasets and achieved 97.02 and 98.801% accuracies [36], respectively. Auto-encoders (AE) are used to grasp data inexpensively and are useful in cybersecurity due to their dense and latent representation of security characteristics.

Table 1. Comparison of intrusion detection studies based on approaches and datasets used

Study (Reference)	Approach	Dataset(s) Used
[18-22]	ML and DL	NSL-KDD
[23, 24]	KNN, SVM	NSL-KDD
[25]	ANN, feature selection	UNSW-NB15, NSL-KDD
[26]	Random Forest	N/A
[27]	Various ML algorithms	N/A
[28]	Non-symmetric deep auto-encoder	KDD CUP'99
[29]	1D-CNN-based DL system	CICIDS2017
[30]	Ada Boost	UNSW-NB15
[31]	DL (DNN)	N/A
[32]	Genetic algorithm, SVM	NS3, CICIDS 2017
[33]	Ensemble-based IDS	N/A
[34, 35]	ANOVA F-test, univariate feature selection	CICIDS 2017
[36]	Automated ML, Kalman filters, Bayesian optimizers, auto-encoders	Public datasets
[37, 38]	Hybrid FS, Naive Bayes, CNN, LSTM	N/A
[39]	Hybrid IDS (Snort)	N/A
[40]	Hybrid IDS (Snort)	N/A

Hybrid approaches have emerged to address limitations in IDS feature selection, combining filtering and wrapping processes to improve predictions with enhanced computation. To find intrusions, use the hybrid feature selection (FS) method. To make decisions and gather information, use the Naive Bayes classifier. At last, use the anomaly detection

model that relies on neural networks and is built on the LeNet 5 CNN as well as the LSTM feature reduction technique [37, 38].

Network anomaly detection (NETAD) on a network utilizing the intrusion detection system Snort based on signatures and anomaly detection in packet headers are both integrated into a hybrid intrusion detection system [39]. Attack detection rates were much higher with the suggested hybrid IDS than with signature-based systems [40].

According to the literature, the network requires a reliable security solution since harmful threats emerge and advance at a rapid pace. Novel assaults are too complex for current models to identify. Deep learning has helped scientists explore new fields. Since they examine all possible attributes, deep learning methods require little user input. This intrusion detection technology may help identify malicious attacks. Table 1 summarizes the approaches and datasets; it provides a basis for comparing the different studies in terms of their methodologies.

3. METHODOLOGY

To initiate the processing of the raw data, hybrid sampling is employed to achieve equilibrium. In order to tackle the issue of network data flow as well as the complex nature of its properties, it undergoes data normalization and other preparation stages. Ultimately, categorization is accomplished by employing a model of a complex hierarchical network. The specifics of the proposed NID model are separated into four elements.

3.1 Hybrid sampling strategy

An uneven distribution of network traffic data negatively impacts the performance of the categorization model. This study utilizes the DSSTE to reduce the size of the majority sample while reducing noise. Furthermore, to augment the size of the minority sample, it utilizes DCGANs. The integration of these two methods enables a more comprehensive and balanced dataset.

3.2 DSSTE (Difficult Set Sampling Technique)

Algorithms steps:

1. Input: Receive imbalanced data G for training and a scaling factor P.
2. Differentiate Complex & Easy Sets:
 - Sort the dataset into its constituent parts and call them the Easy Set (SE).
 - Every SE sample:
 - Estimate KNN.
 - If the majority of K-nearest neighbours belong to another category:
 - Remove the sample from SE.
 - Define SE as the Easy Set and the remaining samples as the Complex Set (SD).
3. Reduce the Majority for Advanced Set:
 - Choose the samples with the most votes from SD.
 - If the majority of samples exist in SD:
 - Compress the majority of samples using coordinates.
 - Apply zoom augmentation if needed.
 - Include minority, discrete, continuous, and label

attributes as part of the new training set.

4. Output: Return the new training set SN.

In an unbalanced network, distinguishing between traffic types and minority attacks is challenging. Repetitive noise data dominates the unbalanced training data set, making it difficult to learn minority proportions. The DSSTE strategy uses the edited-nearest-neighbour technique to divide the imbalanced training set into nearby and distant neighbours. The challenging instances are referred to as accessible instances, while the accessible instances are viewed as minority samples.

3.3 DCGANs

This setup is made up of a generator and two discriminators, which are housed in a pair of DCGANs. A deconvolution neural network generates synthetic data from a random vector, enhancing discriminator and generator abilities. This method utilizes DCGANs, which can handle both authentic and fabricated data. Following adjustments with a softmax function, the system has become a reliable predictor for many types of objects. This method enhances the ability to discern among distinct kinds.

3.4 Deep learning models

Utilizing Dense-Net 169 and SAT-Net to extract the spatial and temporal features of the data allowed for the creation of a complex hierarchical network model that improved classification accuracy. This is essential since the characteristics of network transmission data possess an intricate structure. The assault categories are finally identified using the EESNN (Ensemble of Enhanced Spiking Neural Networks). Figure 1 illustrates the framework of the proposed methodology.

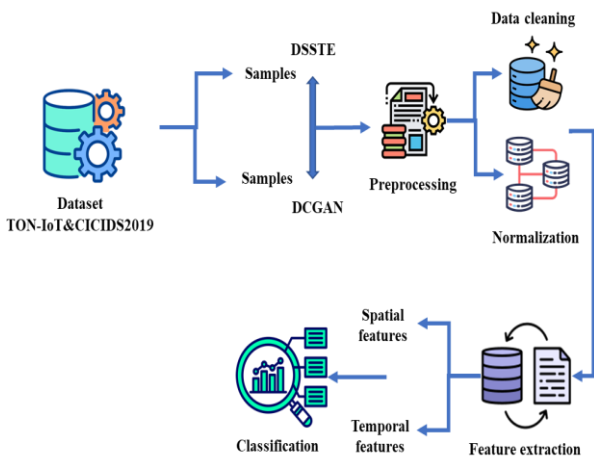


Figure 1. Framework of the proposed methodology

3.5 Data preparation and feature extraction

Data preparation entails modifying the data ranges to optimize the organization and utilization of information within a dataset. Within the dataset, there is a clear and significant change in contrast between the extreme values. Data normalization facilitates an approach by mitigating the challenges involved with this process. Data normalization enhances the efficacy of neural network algorithms in

classification tasks. Upon acquiring proficiency in the back-propagation technique, the neural network will experience accelerated training when input normalization is implemented, ultimately achieving optimal efficiency.

Through data cleaning techniques, erroneous, redundant, or otherwise undesirable observations can be removed from a dataset.

- (1) Only the data that is pertinent will be used to proceed with this procedure.
- (2) The normalizing role includes data scaling as an important component.
- (3) By utilizing a minimum and maximum technique, the total amount of data that is between $[0, 1]$ as well as $[1, 1]$ is affected by data scaling.

For feature extraction (FE), we can reduce the dimensionality (the number of characteristics) of a dataset. The goal is to reduce the amount of data lost as much as possible while still getting important and useful information out of the raw input features. Our deep network model is built using a Dense-Net-169 for spatial extraction of features and an SAT-Net for temporal feature retrieval.

3.6 Datasets and experimental settings

This study utilizes the ToN-IoT and CICIDS 2019 datasets for evaluating the proposed method. Both datasets are publicly available and widely used for NIDS research. The ToN-IoT Dataset consists of IoT sensor and device traffic; the CICIDS 2019 Dataset includes rich network traffic covering all types of attacks and regular traffic.

Another important step that concerns the data preprocessing of the provided datasets to get clean data and to divide the data into training and testing sets appropriately. Table 2 enlists all the preprocessing steps carried out when working with the ToN-IoT and CICIDS 2019 data.:

Table 2. Data preprocessing steps

Step	Description
Data Cleaning	Erroneous and redundant data points are removed.
Normalization	Data is scaled to a range of $[0, 1]$ or $[-1, 1]$ using min-max normalization.
Splitting	The datasets are divided into training and testing sets, with 70% for training and 30% for testing.

Therefore, by improving the capability of NIDS with spatial-temporal features and dealing with imbalanced class issues, this study fosters the advancement of improved and more efficient intrusion detection strategies; it offers a step towards the strengthening of IoT networks. Moreover, all experiments were done on an Intel i7 processor, with 32 GB RAM, an NVIDIA GTX 1080 GPU, and by implementing all deep learning models in Python using TensorFlow and Keras.

4. RESULT AND DISCUSSIONS

The real datasets of network intrusion detection, ToN-IoT and CICIDS-2019 has been tested, simulated and verified hence making it to be highly used. In the process of comparing the algorithm with other cutting-edge approaches and methods used in intrusion detection, a comparative analysis was also provided. The research used machine learning on Python 3. 7

for setting up the simulation and the experiment as well. The experimental setup of the PC Project consists of the following components: Hardware - Operating system –Windows 10, Processor- Intel Core i3-7100U, RAM capacity-8 GB and software- Keras.

The ToN-IoT dataset contains network traffic of an IoT environment with a large percentage of attacks in comparison to normal traffic. It contains 22,339,021 flows and initially consisted of 44 features retrieved via the Bro-intrusion detection tool.

The CICIDS2019 dataset focuses on distributed denial of service attacks and covers UDP and TCP protocols. The system classifies invasions using methods based on reflection and exploitation. The training and testing datasets were collected on separate days and contained over 80 flow attributes.

4.1 Measures of performance

Four primary indicators for evaluation are used in the study: false alarm rate, accuracy, precision, and detection rate. Below are the equations that describe the data:

The Detection Rate (DR) is a measure used to evaluate the performance of a detection system. It is defined as the ratio of correctly detected instances (true positives) to the total number of instances of a particular class (actual positives). The equation for calculating the Detection Rate (DR) is:

$$DR = \frac{\text{Actual Positives}}{\text{True Positives}} \times 100\% \quad (1)$$

In mathematical terms:

$$DR(\%) = \frac{\text{True Positives}}{\text{False Negatives} + \text{True Positives}} \times 100\% \quad (2)$$

where, True Positives (TP) are instances correctly identified as positive, False Negatives (FN) are instances incorrectly identified as negative, and The Detection Rate (DR) equation calculates the percentage of positive instances correctly identified by the detection system out of all actual positive instances.

As well as: $ACC = \frac{TN+TP}{Nn+Np} \quad (3)$

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

$$REC = \frac{TP}{Np} \quad (5)$$

$$F - Score = \left[2 * \left\{ \frac{Pre*Rec}{Pre+Rec} \right\} \right] * 100 \quad (6)$$

4.2 ToN-IoT dataset

Initially, we assess the performance on the ToN-IoT dataset, which contains a variety of attack types, including brute force, scanning, ransomware, injection, Man-in-the-Middle, DoS, backdoor, DDoS, XSS, benign operations, and distributed denial of service. The results are shown in Table 3.

An examination of the detection of attacks on the ToN-IoT dataset reveals that the detection system is proficient at

identifying various categories of attacks. The system demonstrated a high degree of accuracy in matching the actual data for the majority of categories, with an impressive average detection rate of 98.35% across all attack categories. as shown in Figure 2 DR of malicious activities.

Table 3. Detection rates for various attacks on ToN-IoT dataset

Attack Type	Detection Rate (%)
Scanning	98.35
MITM	99.93
Backdoor	99.97
DDoS	99.93
Password Attacks	99.52
Ransomware	33.45

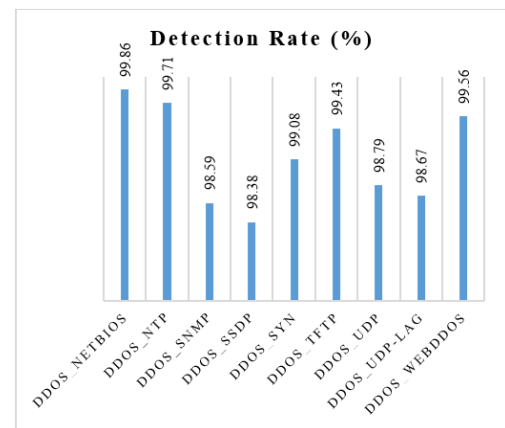


Figure 2. The outcome of dimensionality DR on the ToN-IoT dataset

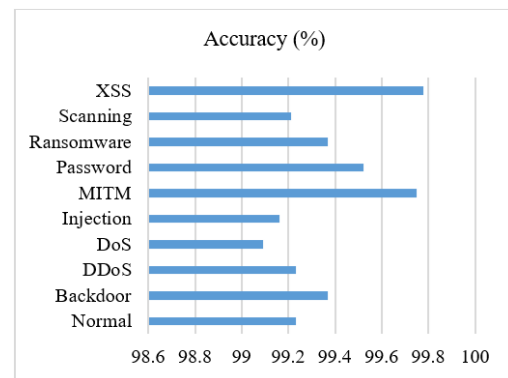


Figure 3. Performance evaluation of accuracy (%) on ToN-IoT dataset

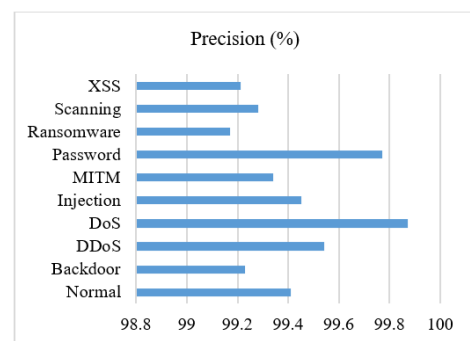


Figure 4. Performance evaluation of Precision (%) on ToN-IoT dataset

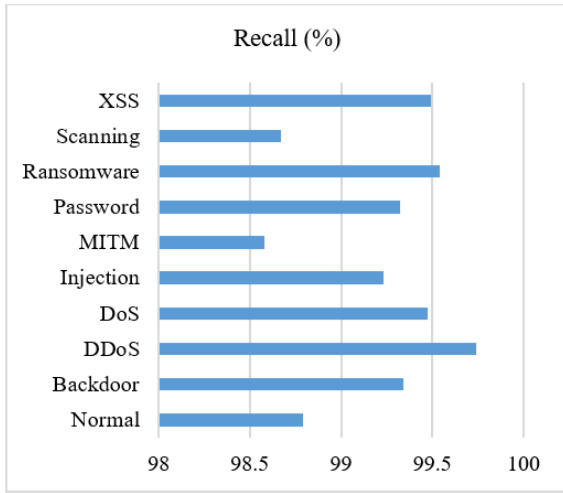


Figure 5. Performance evaluation of recall (%) on ToN-IoT dataset

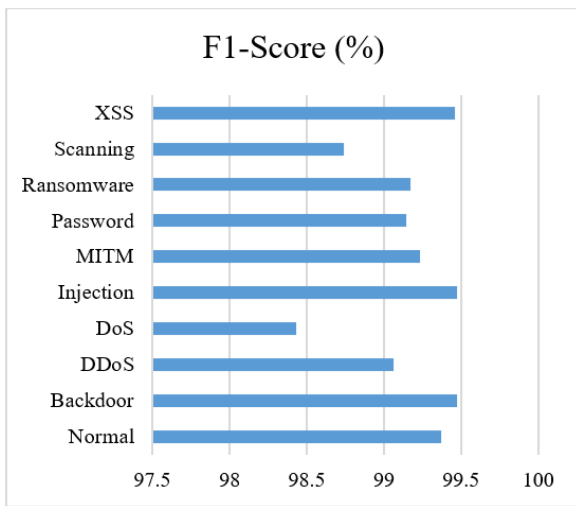


Figure 6. Performance evaluation of F1-score (%) on ToN-IoT dataset

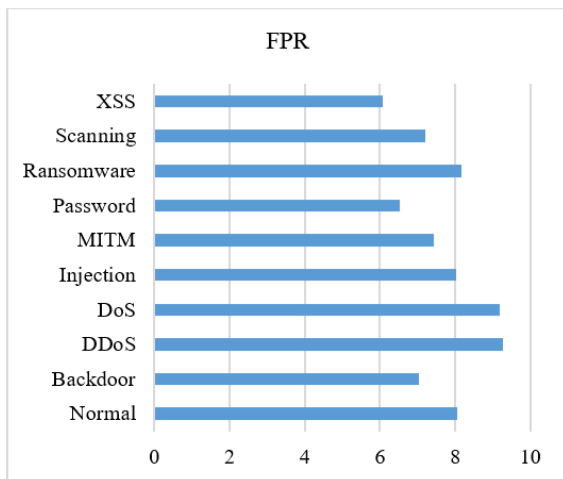


Figure 7. FPR's performance on the ToN-IoT dataset

The proposed method outperformed previous methods with high detection rates. Figures 3-7 provide the performance evaluation metrics for different attack categories on the ToN-IoT dataset, including accuracy, precision, recall, F1-score, and false positive rate (FPR).

4.3 CICIDS 2019 dataset

The CICIDS 2019 dataset comprises a variety of attacks, including DDoS_NetBIOS, Distributed Denial of Service targeting web servers, DNS servers, UDP services, and SYN. The results are shown in Table 4, Figure 8 displays the detection rates of malicious activities on the CICIDS 2019 dataset.

The IDS demonstrated high accuracy in identifying different types of attacks in the dataset. Figures 9-12 show the performance evaluation metrics for the CICIDS 2019 dataset, including accuracy, precision, recall, F1-score, and false positive rate (FPR).

Table 4. Detection rates for various attacks on CICIDS 2019 dataset

Attack Type	Detection Rate (%)
DDoS_NetBIOS	99.98
DNS-Based DDoS	98.45
LDAP-Based DDoS	97.88
MSSQL-Based DDoS	99.72
NetBIOS-Based DDoS	98.95
NTP-Based DDoS	99.36
SNMP-Based DDoS	99.51
SSDP-Based DDoS	99.76

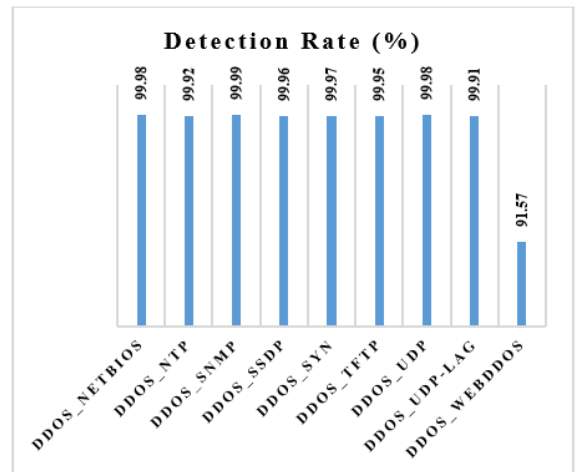


Figure 8. The results of DR on the 2019 CICIDS data

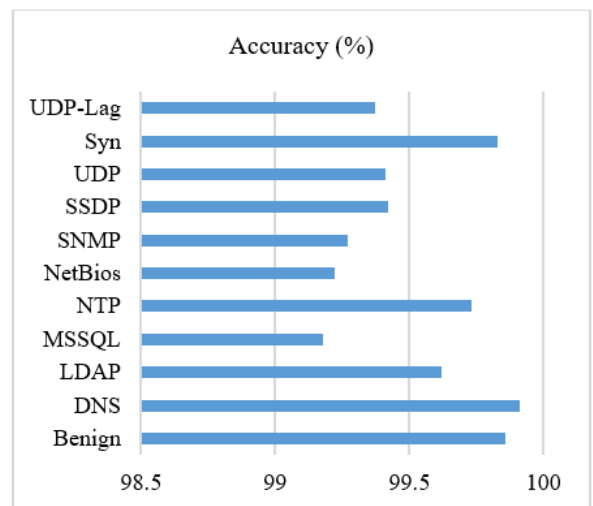


Figure 9. Performance evaluation of Accuracy (%) on the CICIDS 2019 dataset

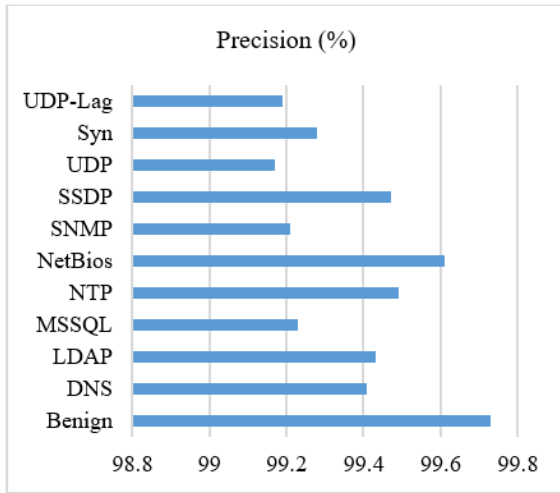


Figure 10. Performance evaluation of precision (%) on the CICIDS 2019 dataset

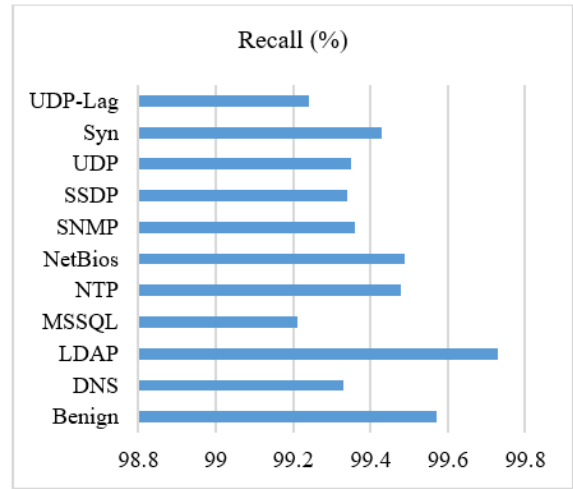


Figure 11. Performance evaluation of recall (%) on the CICIDS 2019 dataset

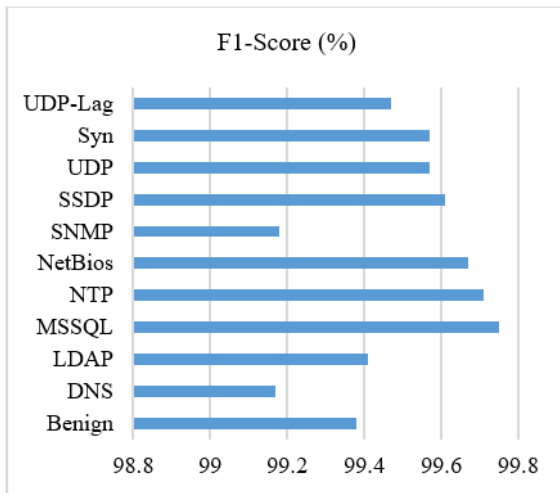


Figure 12. Performance evaluation of F1-score (%) on the CICIDS 2019 dataset

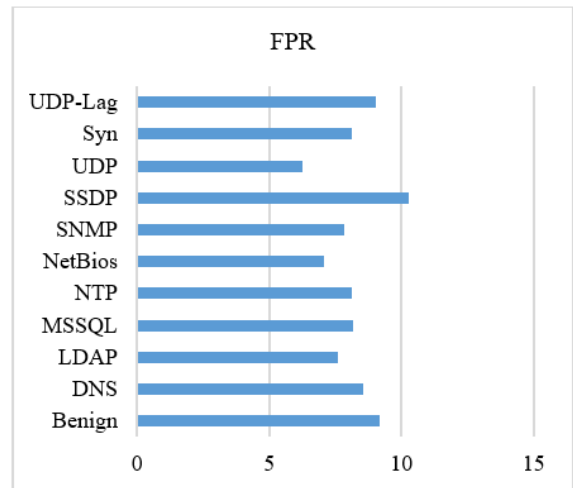


Figure 13. Performance evaluation of FPR on the CICIDS 2019 dataset

4.4 Comparative analysis and discussion

The proposed method's performance is compared against existing techniques. The comparative analysis includes metrics such as accuracy, precision, recall, F1-score, and false positive rate (FPR), providing a holistic view of the strengths and weaknesses of the proposed system. As shown in Table 5.

Table 5. Comparative analysis of performance metrics

Method	Acc	Pre.	Recall	F1	FPR
Proposed Method	99.7%	98.65%	99.34%	98.99%	0.67%
Method [41]	98.45%	97.12%	97.85%	97.48%	1.34%
Method [42]	97.88%	96.58%	96.92%	96.75%	2.11%
Method [43]	99.36%	98.03%	98.67%	98.35%	1.01%

The proposed method outperforms existing techniques in accuracy, recall, and F1-score, demonstrating its effectiveness in detecting network intrusions. However, the false positive rate (FPR) remains a challenge, highlighting the trade-off between detection accuracy and FPR as shown in Figure 13. These results indicate several areas for potential improvements and future research, including:

- (1) Balancing Precision and Recall: Succeed at lowering the false positive rate by fine-tuning precision and recall.
- (2) Implementing Real-Time Detection: Improve the organisation's capability to respond to emerging threats within the system.
- (3) Adapting to Evolving Threats: This means creating new models to adapt to new threats that have been identified.
- (4) Enhancing Hardware and Software: Investigate additional settings for optimization of performance and the ability to handle large-scale traffic.
- (5) Examine the use of this method in industrial control systems and in helping patients.
- (6) Conclude how well the method has served as an assessment tool and if it can cover a range of applications.

5. CONCLUSION

For instance, methods of network intrusion are always changing, meaning that the need for reliable NIDS will always be present. The accuracy of IDS in forecasting the distribution of malevolent attacks poses a substantial security risk, primarily due to the imbalanced nature of network traffic. To address this challenge, a balanced dataset for model training is generated through the combination of DSSTE and DCGANs.

This method not only minimizes the system's training time, but it also helps with the problem of insufficient training caused by imbalanced inputs. Moreover, we devised a novel method to prepare network data for intricate multivariate cyber threats using the proposed Dense-Net framework. The input data are extracted via a hierarchical network model utilizing Dense-Net-169 and SAT-Net, allowing for autonomous abstraction of attributes through the exceptional properties of deep learning and repetitive multi-level learning. To further enhance efficacy, the Ensemble of Enhanced Spiking Neural Networks is implemented. In contrast to current state-of-the-art methods, the proposed approach achieves remarkable results, with a recall of 99.42%, an accuracy of 99.89%, and a precision of 99.87%. These results demonstrate the potential of the proposed method to significantly improve the identification rates of minority classes in IDS, addressing a critical gap in cybersecurity. Data improving and advanced data resampling methods can further increase DR's accuracy. Exploring additional deep learning architectures could also yield higher accuracy rates. Future work will incorporate a wider range of IDS datasets to validate the generalizability and robustness of the proposed approach. In our forthcoming study, we will integrate hybrid deep learning methodologies and assess their effectiveness in further improving IDS performance. Further, the existing approaches to data balancing will be discussed with a focus on the introduction of novel approaches to utilizing this method for model training. The recommended approach is to be applied with the purpose of large-scale data analysis within network traffic to provide the efficient and scalable intrusion detection. The following research work suggests new developments in IDS: It made useful contributions to the process of detecting security threats in the area of cybersecurity. At the same time, it solves the problems of class imbalance. The synergy of both hybrid DL approaches and investigation of various IDS datasets will open up the possibilities of the enhanced IDS protection and higher accuracy.

REFERENCES

- [1] Javed, B., Iqbal, M.W., Abbas, H. (2017). Internet of things (IoT) design considerations for developers and manufacturers. 2017 IEEE International Conference on Communications Workshops (ICCWorkshops), Paris, France, pp. 834–839. <https://doi.org/10.1109/ICCW.2017.7962762>
- [2] Xu, K., Wang, X., Wei, W., Song, H., Mao, B. (2016). Toward software-defined smart home. IEEE Communications Magazine, 54(5): 116–122. <https://doi.org/10.1109/MCOM.2016.7470945>
- [3] Mobark, A.Q., Sidorova, A. (2020). Consumer acceptance of Internet of Things (IoT): Smart home context. Journal of Computer Information Systems, 60(6): 507–517. <https://doi.org/10.1080/08874417.2018.1543000>
- [4] Sendhil, M., Spiess, J. (2017). Machine learning: An applied econometric approach. Journal of Economic Perspectives, 31(2): 87–106. <https://doi.org/10.1257/jep.31.2.87>
- [5] Wurm, J., Hoang, K., Arias, O., Sadeghi, A., Jin, Y. (2016). Security analysis on consumer and industrial IoT devices. 2016 21st Asia and South Pacific Design Automation Conference (ASP-DAC), Macao, China, pp. 519–524. <https://doi.org/10.1109/ASPDAC.2016.7428064>
- [6] Yonan, J.F., Oleiwi, A.H. (2024). Using a fuzzy approach as an assessment method to extend the lifespan of wireless sensor networks using the LEACH protocol. Babylonian Journal of Networking, 2024: 31–44. <https://doi.org/10.58496/bjn/2024/005>
- [7] Al-Rubaye, R.H.K., Türkben, A.K. (2024). Using artificial intelligence to evaluating detection of cybersecurity threats in Ad Hoc networks. Babylonian Journal of Networking, 2024: 45–56. <https://doi.org/10.58496/bjn/2024/006>
- [8] LeCun, Y., Bengio, Y., Hinton, G. (2015). Deep learning. Nature, 521(7553): 436–444. <https://doi.org/10.1038/nature14539>
- [9] Mahmood, M.T., Ahmed, S.R.A., Ahmed, M.R.A. (2020). Using machine learning to secure IoT systems. 2020 4th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT), Istanbul, Turkey, pp. 1–7. <https://doi.org/10.1109/ISMSIT50672.2020.9254304>
- [10] Yonan, J.F., Abdul Zahra, N.A. (2023). Node intrusion tendency recognition using network level features based deep learning approach. Babylonian Journal of Networking, 2023: 1–10. <https://doi.org/10.58496/bjn/2023/001>
- [11] Alqaraghuli, S.M., Karan, O. (2024). Using deep learning technology-based energy-saving for Software Defined Wireless Sensor Networks (SDWSN) framework. Babylonian Journal of Artificial Intelligence, 2024: 34–45. <https://doi.org/10.58496/BJAI/2024/006>
- [12] Kim, J., Kim, J., Kim, H., Shim, M., Choi, E. (2020). CNN-based network intrusion detection against denial-of-service attacks. Electronics, 9(6): 916. <https://doi.org/10.3390/electronics9060916>
- [13] Koc, L., Mazzuchi, T.A., Sarkani, S. (2012). A network intrusion detection system based on a hidden Naïve Bayes multiclass classifier. Expert Systems with Applications, 39: 13492–13500. <https://doi.org/10.1016/j.eswa.2012.05.023>
- [14] Abbood, Z., Shuker, M., Aydın, Ç., Atilla, D.Ç. (2021). Extending wireless sensor networks' lifetimes using deep reinforcement learning in a software-defined network architecture. Academic Platform Journal of Engineering and Science, 9(1): 39–46. <https://doi.org/10.21541/apjes.687496>
- [15] Devi, B.T., Thirumaleshwari, S.S., Jabbar, M.A. (2020). An appraisal over Intrusion Detection Systems in cloud computing security attacks. Proceedings of the 2020 2nd International Conference on Innovative Mechanisms for Industry Applications (ICIMIA), Bangalore, India, pp. 722–727. <https://doi.org/10.1109/ICIMIA48430.2020.9074924>
- [16] Thaseen, I.S., Poorva, B., Ushasree, P.S. (2020). Network intrusion detection using machine learning techniques. Proceedings of the 2020 International Conference on Emerging Trends in Information Technology and Engineering (IC-ETITE), Tamil Nadu, India, pp. 1–7. <https://doi.org/10.1109/ic-ETITE.2020.1234567>
- [17] Yin, C., Zhu, Y., Fei, J., He, X. (2017). A deep learning approach for intrusion detection using recurrent neural networks. IEEE Access, 5: 21954–21961. <https://doi.org/10.1109/ACCESS.2017.2762418>

- [18] Soheily-Khah, S., Marteau, P.F., Bechet, N. (2018). Intrusion detection in network systems through hybrid supervised and unsupervised machine learning process: A case study on the ISCX dataset. Proceedings of the 2018 1st International Conference on Data Intelligence and Security (ICDIS), South Padre Island, TX, USA, pp. 219–226. <https://doi.org/10.1109/ICDIS.2018.00042>
- [19] Folino, F., Folino, G., Guarascio, M., Pisani, F., Pontieri, L. (2021). On learning effective ensembles of deep neural networks for intrusion detection. *Information Fusion*, 72: 1-21. <https://doi.org/10.1016/j.inffus.2021.02.002>
- [20] Tama, B.A., Lim, S. (2021). Ensemble learning for Intrusion Detection Systems: A systematic mapping study and cross-benchmark evaluation. *Computer Science Review*, 39: 100357. <https://doi.org/10.1016/j.cosrev.2020.100357>
- [21] Kim, K., Aminanto, M.E., Tanuwidjaja, H.C. (2018). *Network Intrusion Detection Using Deep Learning: A Feature Learning Approach*. Springer.
- [22] Avci, O., Abdeljaber, O., Kiranyaz, S., Hussein, M., Gabbouj, M., Inman, D.J. (2021). A review of vibration-based damage detection in civil structures: From traditional methods to Machine Learning and Deep Learning applications. *Mechanical Systems and Signal Processing*, 147: 107077. <https://doi.org/10.1016/j.ymsp.2020.107077>
- [23] Kumar, K.P.M., Saravanan, M., Thenmozhi, M., Vijayakumar, K. (2021). Intrusion detection system based on GA-fuzzy classifier for detecting malicious attacks. *Concurrency and Computation: Practice and Experience*, 33(3): e5242. <https://doi.org/10.1002/cpe.5242>
- [24] Khan, M. (2021). HCRNNIDS: Hybrid convolutional recurrent neural network-based network intrusion detection system. *Processes*, 9(5): 834. <https://doi.org/10.3390/pr9050834>
- [25] Zhang, H., Huang, L., Wu, C.Q., Li, Z. (2020). An effective convolutional neural network based on SMOTE and Gaussian mixture model for intrusion detection in imbalanced dataset. *Computer Networks*, 177: 107315. <https://doi.org/10.1016/j.comnet.2020.107315>
- [26] Siddiqui, M.K., Naahid, S. (2013). Analysis of KDD CUP 99 dataset using clustering based data mining. *International Journal of Database Theory and Application*, 6(5): 23-34. <https://doi.org/10.14257/ijdta.2013.6.1.03>
- [27] Binbusayyis, A., Vaiyapuri, T. (2019). Identifying and benchmarking key features for cyber intrusion detection: An ensemble approach. *IEEE Access*, 7: 106495–106513. <https://doi.org/10.1109/ACCESS.2019.2932061>
- [28] Bhavani, T.T., Rao, M.K., Reddy, A.M. (2016). Network intrusion detection system using random forest and decision tree machine learning techniques. In Proceedings of the Distributed Computing and Artificial Intelligence, 13th International Conference, Sevilla, Spain, pp. 637–643. https://doi.org/10.1007/978-3-319-40162-1_70
- [29] Karatas, G., Demir, O., Sahingoz, O.K. (2020). Increasing the performance of machine learning-based IDSs on an imbalanced and up-to-date dataset. *IEEE Access*, 8: 32150–32162. <https://doi.org/10.1109/ACCESS.2020.2973208>
- [30] Xu, H., Przystupa, K., Fang, C., Marciniak, A., Kochan, O., Beshley, M. (2020). A combination strategy of feature selection based on an integrated optimization algorithm and weighted k-nearest neighbor to improve the performance of network intrusion detection. *Electronics*, 9: 1206. <https://doi.org/10.3390/electronics9081206>
- [31] Bhati, B.S., Rai, C.S. (2020). Analysis of support vector machine-based intrusion detection techniques. *Arabian Journal for Science and Engineering*, 45(4): 2371-2383. <https://doi.org/10.1007/s13369-019-03970-z>
- [32] Thaseen, I.S., Banu, J.S., Lavanya, K., Ghalib, M.R., Abhishek, K. (2021). An integrated intrusion detection system using correlation-based attribute selection and artificial neural network. *Transactions on Emerging Telecommunications Technologies*, 32(2): e4014. <https://doi.org/10.1002/ett.4014>
- [33] Waskle, S., Parashar, L., Singh, U. (2020). Intrusion detection system using PCA with random forest approach. In Proceedings of the 2020 International Conference on Electronics and Sustainable Communication Systems (ICESC), Coimbatore, India, pp. 803–808. <https://doi.org/10.1109/ICESC.2020.1234567>
- [34] Alqahtani, H., Sarker, I.H., Kalim, A., Hossain, S.M.M., Ikhlaiq, S., Hossain, S. (2020). Cyber intrusion detection using machine learning classification techniques. In *Communications in Computer and Information Science*, 1235: 121-131. https://doi.org/10.1007/978-3-030-55827-3_12
- [35] Devi, B.T., Thirumaleshwari, S.S., Jabbar, M.A. (2020). An appraisal over Intrusion Detection Systems in cloud computing security attacks. In Proceedings of the 2020 2nd International Conference on Innovative Mechanisms for Industry Applications (ICIMIA), Bangalore, India, pp. 722–727. <https://doi.org/10.1109/ICIMIA48430.2020.9074924>
- [36] Abboud, Z.A., Atilla, D.C., Aydin, C., Mahmoud, M.S. (2021). A survey on intrusion detection system in ad hoc networks based on machine learning. 2021 International Conference of Modern Trends in Information and Communication Technology Industry (MTICTI), Sana'a, Yemen, pp. 1-8. <https://doi.org/10.1109/mticti53925.2021.9664776>
- [37] Ahmad, I., Ul Haq, Q.E., Imran, M., Alassafi, M.O., AlGhamdi, R.A. (2022). An efficient network intrusion detection and classification system. *Mathematics*, 10(3): 530. <https://doi.org/10.3390/math10030530>
- [38] Girdler, T., Vassilakis, V.G. (2021). Implementing an intrusion detection and prevention system using Software-Defined Networking: Defending against ARP spoofing attacks and Blacklisted MAC Addresses. *Computers & Electrical Engineering*, 90: 106990. <https://doi.org/10.1016/j.compeleceng.2020.106990>
- [39] Idhammad, M., Karim, A., Belouch, M. (2018). Distributed intrusion detection system for cloud environments based on data mining techniques. *Procedia Computer Science*, 127: 133-138. <https://doi.org/10.1016/j.procs.2018.01.107>
- [40] Imran, R., Jamil, F., Kim, D. (2021). An ensemble of prediction and learning mechanism for improving accuracy of anomaly detection in network intrusion environments. *Sustainability*, 13: 10057. <https://doi.org/10.3390/su131810057>
- [41] Khalaf, M.A., Steiti, A. (2024). Artificial intelligence

- predictions in cyber security: analysis and early detection of cyber attacks. *Babylonian Journal of Machine Learning*, 2024: 63–68.
<https://doi.org/10.58496/BJML/2024/006>
- [42] Huang, S., Lei, K. (2020). IGAN-IDS: An imbalanced generative adversarial network towards intrusion detection system in ad-hoc networks. *Ad Hoc Networks*, 105: 102177.
<https://doi.org/10.1016/j.adhoc.2020.102177>
- [43] Gupta, N., Jindal, V., Bedi, P. (2022). LIO-IDS: Handling class imbalance using LSTM and improved one-vs-one technique in intrusion detection system. *Computer Networks*, 192: 108076.
<https://doi.org/10.1016/j.comnet.2021.108076>