A Comprehensive Study on Deep Learning Techniques for IoT Security

Abdeslem Blali [1], Souhayla Dargaoui [1,*], Mourade Azrour [1], Azidine Guezzaz [2], Abdulatif Alabdulatif [3], and Fatima Amounas [1]

[1] IMIA Laboratory, MSIA Team, Faculty of sciences and Techniques, Moulay Ismail University of Meknes, Errachidia, Morocco
[2] Department of Computer Science and Mathematics, Higher School of Technology Essaouira, Cadi Ayyad University, Essaouira, Morocco
[3] Department of Computer Science, College of Computer, Qassim University, Buraydah, Saudi Arabia
Email: ab.blali@edu.umi.ac.ma (A.B.); s.dargaoui@edu.umi.ac.ma (S.D.); mo.azrour@umi.ac.ma (M.A.); a.guezzaz@uca.ma (A.G.); ab.alabdulatif@qu.edu.sa (A.A.), f.amounas@umi.ac.ma (F.A.)
*Corresponding author

***Abstract*—The present paper looks at the security problems caused by the fast growth of the Internet of Things (IoT) in areas like industry, healthcare, and agriculture. As IoT systems become more common, they face more threats from cyberattacks like Brute Force, Denial of Service (DoS), Botnets, and so ones. To deal with these security issues, we studied recent papers to review different intrusion detection systems made for IoT. The goal was to see how well they work and find ways to make them better. Hence, we have selected 63 relevant papers among 1200 find papers. These selected papers were published between 2020 and 2024. Our study shows that deep learning-based intrusion detection systems can improve the manner how online threats are detect. These systems, especially when they use neural networks, are better at spotting and reacting to harmful activities. Combining machine learning with Intrusion Detection Systems (IDS) seems to help boost the security of internet of things networks, offering stronger protection against cyber-attacks. One of the best algorithms we found was the combination of a Convolutional Neural Network (CNN) and a Long Short-Term Memory (LSTM) network. This deep learning model showed very high accuracy in protecting IoT networks, especially when tested with different datasets. This proves that using advanced algorithms is important to keep up with the growing challenges of cyber-threats targeting IoT systems.**

***Keywords*—Internet of Things (IoT), security, intrusion detection, Intrusion Detection Systems (IDS), deep learning, neural networks, Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM)**

I. Introduction

The Internet of Things (IoT), projected to exceed 30 billion active device connections globally by 2025, presents an expansive attack surface [1]. In fact, IoT technologies have changed the way devices interconnect and share data [2]. This transformation has provided significant benefits in various fields like healthcare, smart home, and agriculture [3].

However, it brings with it different security issues, such as data violations, non-authorized access and system viabilities. To address these challenges, strong security solutions are required. Hence, key security features include encrypting data to block unauthorized access, authorizing devices to guarantee their legitimacy, and keeping software regularly updated to correct any vulnerability. These strategies are vital for protecting IoT devices and ensuring the stability and safety of interconnected systems [4].

The first important solution is data encryption. When IoT devices communicate with each other, they often exchange sensitive information. Encryption makes this information unreadable to unauthorized people by converting it into a complex code. Hence, IoT system uses encryption keys that are needed to decrypt the data. So, even if someone intercepts the information, it remains protected and impossible to understand [5].

Device authentication and identity management are also essential in an IoT network. Each device must prove it is authorized to access the network. This can be done with complex passwords or more advanced systems like multi-factor authentication. These algorithms ensure that only legitimate users and devices can interact with the network, reducing the risk of intrusions by malicious individuals [6].

Finally, regular software updates are another important solution for the security of IoT devices. IoT devices can have vulnerabilities in their programs that hackers can exploit. Additionally, Intrusion Detection Systems (IDS) play a crucial role in IoT security by monitoring network traffic for signs of malicious activity. IDS can quickly detect and alert users to potential cyberattacks, providing an extra layer of defense that complements software updates and other security measures [7]. IDS play a key

Manuscript received April 21, 2025; revised July 21, 2025; accepted August 15, 2025; published December 18, 2025.

role in protecting against different risks [1]. However, the traditional methods used for design IDSs face major issues in various IoT networks. However, the fact that they are not sufficiently adaptable to the variety of different IoT technologies and protocols decreases the efficiency

In the past years, researchers have used both Machine Learning (ML) and Deep Learning (DL) algorithms to improve IDS efficiency [8–17]. Hence, ML allows to the systems to automatically gather valuable information from large amounts of data. Besides, with sufficient training dataset, ML-based IDS can successfully detects threats. Moreover, these models can be build and use without investigating hard efforts. On the other hand, DL uses models that are designed to work like the human brain's neural networks. These models help solve problems related to IoT devices by understanding data and applying it in various fields. In addition, DL algorithms are generally more effective than ML, especially when dealing with large datasets. However, IDS still face challenges in quickly detecting intrusions because of the growing network traffic and security risks.

According to Ref. [2], Zipperle *et al.* performed an in-depth review of existing research on provenance-based intrusion detection systems. The survey categorized different types of provenance-based intrusion detection systems and highlighted the importance of utilizing real-world datasets. It also discussed key aspects such as data collection processes, graph summarization techniques, and methods for detecting intrusions within the provenance-based intrusion detection systems framework.

As shown in Ref. [3], a comprehensive assessment of ML-based IDS was presented. The review thoroughly examined various ML techniques, outlining their respective advantages and limitations in the context of IDS.

As reported in Ref. [4], a systematic review focused on the application of Natural Language Processing (NLP) methods in Host-based Intrusion Detection System (HIDS). The study classified the NLP techniques used in HIDS, and also discussed relevant datasets and evaluation metrics employed in NLP-driven HIDS research.

This paper focuses on reviewing recent papers that address IDS using deep learning and explores how these systems can improve the security of IoT devices. It also looks at the limitations of each system. The main research questions include the algorithms used in the recent years, the deep learning algorithms chosen, the complexity of each proposal, and the types of attacks included in the datasets used to develop deep learning-based IDS.

In fact, the paper's novelty lies in its focused review of recent deep learning-based Intrusion Detection Systems (IDS) tailored for IoT security, highlighting advancements in algorithm selection, system complexity, and dataset utilization. Unlike previous surveys that broadly addressed IDS or machine learning approaches, this work critically examines the specific deep learning models applied to IoT environments, their effectiveness, and the limitations they face.

The rest of the paper is organised as following. In the Section II, we give some background details about IoT systems and IoT threats. Section III presents the adopted methodology for section the reviewed papers. Section IV focuses on intrusion detection systems deployed in IoT environments, detailing the techniques and algorithms used. Section V concludes with a summary of the findings and possible future directions for improving IoT security.

II. BACKGROUND

A. Internet of Things Conception

The IoT concept originated in 1999 at the Massachusetts Institute of Technology through networks utilizing Radio Frequency Identification (RFID) technology. Initially, the system's primary functions included data collection, processing, transmission, and application. Although not all "Things" are necessarily connected to the internet, IoT is generally understood as a vast network of objects, sensors, and actuators designed for specific purposes. This broadens the idea to include both isolated networks and those connected via the internet, collectively known as the Network of Things. As explained earlier, IoT involves interconnected sensors and actuators that communicate autonomously to generate and exchange data for meaningful functions. Its main role is to gather information from the physical environment and deliver services based on data analysis or user requests. In IoT systems, digital entities are directly linked to physical objects that interact and collaborate to accomplish various tasks. IoT finds applications across various domains beyond research and industry, including smart grids, e-health, smart homes, environmental monitoring, and smart cities. Regarding architecture, IoT remains somewhat ambiguous, with no universal standard. The most common is a three-layer model comprising the Application, Network, and Perception layers, but this simplistic structure is limited and cannot fully address the complexities of modern IoT applications.

B. IoT Threats

The main threats to IoT systems encompass a range of security vulnerabilities and risks that can compromise data integrity, privacy, and system functionality. Key threats include unauthorized access and device hijacking, which can allow hackers to take control of IoT devices or eavesdrop on sensitive data. Data breaches and interception pose significant risks, as sensitive information transmitted across IoT networks can be intercepted or tampered with. Malicious software and firmware updates can exploit vulnerabilities to launch cyberattacks or disable devices. Additionally, weak authentication mechanisms and poor encryption practices increase the likelihood of intrusion. Distributed Denial of Service (DDoS) attacks, often leveraging compromised IoT devices, threaten network availability and disrupt service. Physical attacks or tampering with devices can also lead to data corruption or device malfunction. Overall, these threats highlight the critical need for robust security measures tailored to the unique challenges of IoT environments.

III. Materials and Methods

This section describes the comprehensive methodology used to conduct the systematic review of Deep Learning-based intrusion detection approaches adopted recently in IoT systems. The methodology followed is based on the guidelines described in the Preferred Reporting Items for Systematic Reviews and Meta-Analyses (PRISMA) method (Fig. 1).

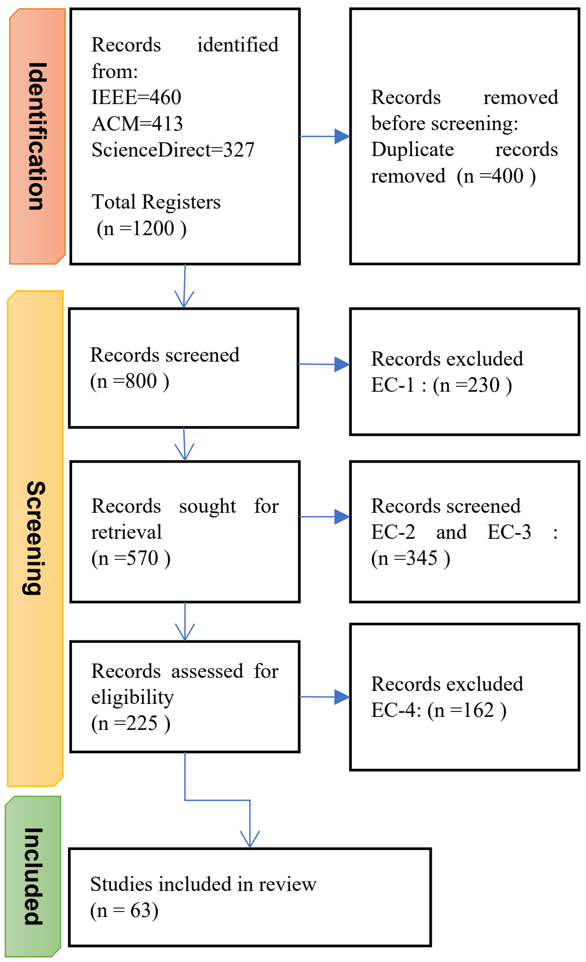

Fig. 1. PRISMA flowshart.

A. Search Strategy

Our study was carried out over the period 2020–2024, as IoT IDS has recently received significant attention. Research conducted over the past five years presents emerging technologies and current trends in DL-based IDS effectiveness, helping to strengthen understanding of the state of the art. In order to identify the paper that we can adopted in our analysis, we have done research in various databases like: Google Scholar (https://scholar.google.com/), ACM Digital Library (http://dl.acm.org), IEEE eXplore (http://ieeexplore.ieee.org), and ScienceDirect (https://www.sciencedirect.com). In this study, Springer dataset was excluded because the majority of documents within it (Springer) are either book chapters or conference papers. Additionally, most of the journal papers available through Springer are not accessible freely, which limited their inclusion in the study. These accessibility issues and the document types contributed to the decision to omit the Springer dataset from our analysis.

Table I delineates the criteria for selecting relevant manuscripts focused on IDS, emphasizing the inclusion of specific keywords such as "Intrusion Detection", "Internet of Things", and "Deep Learning". The selection process mandates that only peer-reviewed, full-text articles written in English and available electronically are considered. Non-English publications, articles that are not in electronic format, and studies published outside the designated time frame are excluded. These stringent criteria aim to ensure the selection of high-quality, pertinent research relevant to the study.

TABLE I. Criteria for Inclusion and Exclusion of Studied Papers

Inclusion Criteria (IC)	Exclusion Criteria (EC)
IC-1: Studies must pertain to Intrusion Detection System. These keywords are central to the research's aim, such as "IoT", "Internet of Things", "IoT system", "intrusion detection", "anomaly detection", "deep learning" **IC-2:** Only peer-reviewed papers that align with the objectives outlined in IC-1 will be considered. **IC-3:** Full-text articles must comply with the criteria set forth in IC-1 and IC-2.	**EC-1:** Papers which are not published in English are excluded. **EC-2:** Papers that are not available in electronic file format are not included. **EC-3:** Review studies and non relevant. The main topic is not IDS **EC-4:** Papers that are not published between studied period (2019-2024)

In this research, we have determined the specific search words to find the appropriate papers. Hence, three fields were careful examined in our search which title, abstract, and keywords. The search expressions where created based numerous keywords conforming to the suitable conditions and combined with Boolean operators (AND, OR, NOT). The search expressions used are:

- IoT: "IoT", "Internet of Things", "IoT system";
- Intrusion detection: "intrusion detection", "anomaly detection";
- Deep learning: "artificial intelligence", "deep learning", "Neural Network".

Therefore, the used query phrase is ("IoT" OR "Internet of Things" OR "IoT system") AND ("intrusion detection" OR "anomaly detection") AND ("artificial intelligence" OR "deep learning" AND "Neural Network") NOT ("machine learning")

B. Study Selection

After gathering various paper find in queries step, we have done the selection of the paper we will analysis. The main goal of this step is to eliminate studies that are not relevant to our analysis. Hence, this goal is accomplished through a filtering process that usually begins with the verification of study titles, abstracts, and whole texts. Once this pre-selection has been completed, we can obtain the remaining studies to assess their relevance. Hereafter, the complete flowchart of the selection process, comprising identification, selection, qualification and inclusion, is illustrated in Fig. 1.

IV. Intrusion Detection Systems Employed in the IoT Environment

A. Analysis of Adopted Software

The tools and technologies used in deep learning and embedded computing are very important because they help researchers and developers create and improve innovative projects (Fig. 2.). Tools like Python, MATLAB, and simulation platforms such as Cooja provide environments for programming, modeling, and testing ideas. Libraries like TensorFlow and Keras make it easier to handle complex tasks like building and training deep learning models. These tools save time, encourage teamwork, and provide a common way for researchers to work. Understanding their role helps us see how they influence research and practical applications in these fields [18].

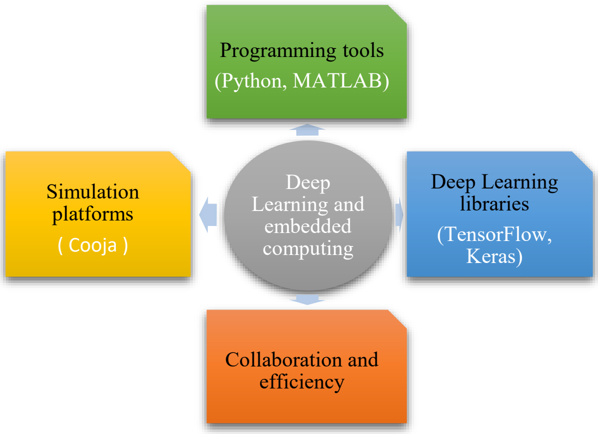

Fig. 2. Key tools and their roles in deep learning and embedded computing.

An organized view of the various technologies and software associated with deep learning and embedded computing is presented. It highlights the tools used in creating and implementing projects in these fields, including development environments such as Python, MATLAB, and Cooja, as well as widely used libraries like TensorFlow, Keras, and Scikit. The inclusion of checkmarks indicates the application of these tools in specific studies, offering a clear representation of the technological choices made during different research or development stages.

The preferences and orientations of developers and researchers in their exploration of artificial intelligence and embedded computing are revealed [19]. The intersections of technologies such as Python, TensorFlow, and Keras suggest a combined use of these popular tools to create deep learning models. Similarly, the inclusion of Raspberry Pi and Cooja highlights the interest in embedded solutions and experimentation in real or simulated environments.

Python is the most used programming language in the reviewed papers. It appears 16 times, representing 50% of the analyzed papers [20]. This frequent use suggests that Python is often paired with frameworks and platforms like Keras or TensorFlow. In contrast, MATLAB is used less frequently, appearing in only three papers.

Finally, it is clear that researchers prefer to conduct tests and simulations with simulators like Cooja. It is rare for them to deploy their frameworks on embedded systems such as the Raspberry Pi, which is mentioned only in [21].

B. Analysis of the Used Datasets

Deep learning-based IDS require datasets to evaluate intrusions effectively. Creating suitable data for model training is crucial yet challenging, as it involves identifying labeled normal and abnormal communications and other features such as IP addresses. However, some datasets used for network traffic analysis remain unavailable to the public due to security concerns. Various datasets utilized in the reviewed studies are described, providing insights into their characteristics and applications [19].

As depicted in Table I, the datasets used for intrusion detection show considerable diversity in terms of IoT specificity, publication years, characteristics, classes, and volumes of normal and attack records. Notable datasets include Telecommunications and Operational Networks-Internet of Things dataset (TON-IOT) (2020) with 44 features and 161,043 attack records, and Botnet Internet of Things dataset (BoT-IoT) (2018) with 46 features, 5 classes, 477 normal records, and 3,668,045 attack records. Other widely used datasets like Network Security Laboratory Knowledge Discovery in Databases dataset (NSL-KDD) (1998) and University of New South Wales Network-Based 2015 dataset (UNSW-NB15) (2015) are not specific to IoT but remain popular due to their rich feature sets and classes.

As we can remake in the Table A1 (in Appendix), more than 20 datasets are published between 2018 and 2023. Hence, these datasets comprises various features that vary between 12 and 88 features. Furthermore, the total number of included attacks exceeds thousands, ranging from 2046 to 45588384 attacks. It is also clear that some recent datasets, like Network Flows-Communications Security Establishment and Canadian Institute for Cybersecurity Intrusion Detection System 2018 dataset (NF-CSE-CIC-IDS2018) (2020) and Network Flows-University of Queensland Network Intrusion Detection System dataset (NF-UQ-NIDS) (2020), comprise millions of records. Such diversity provides the opportunity for researchers to develop and implement IDS solutions that can handle a vast range of different scenarios in IoT environments.

C. Adopted Deep Learning Algorithms

In our days, the increasing number of IoT equipment has raised the importance of strong security measures, particularly for detecting intrusions. By the way, numerous different approaches and algorithms have been developed or tested to address the particular challenges of today's IoT systems, focusing on intrusion detection in a variety of datasets. The reviewed studies indicate a diverse range of methods, including Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM), and even more complex hybrid models and combined systems. Such algorithms are employed on datasets including Communications Security Establishment and Canadian

Institute for Cybersecurity-Intrusion Detection System 2018 dataset (CSE-CIC-IDS2018), BoT-IoT and TON-IOT, illustrating the variety and progress of approaches to improving intrusion detection in IoT systems. In this section, we review the main results of these studies and highlight the different approaches that are shaping the future of IoT security [20].

As depicted in Table A2 (in Appendix), various algorithms are used for implementing IDS in IoT, including CNN, LSTM, Deep Neural Network (DNN), and their combinations. Hence, CNN is used in CSE-CIC-IDS2018, while hybrid models like CNN-LSTM is used in BoT-IoT and Canadian Institute for Cybersecurity Intrusion Detection System 2017 dataset (CICIDS2017). Advanced algorithms like Autoencoder (AE) and Recurrent Neural Network-Gated Recurrent Unit (RNN-GRU) are also used. AE is used in CSE-CIC-IDS2018 and RNN-GRU is used in TON-IOT. Integrated models like Deep Integrated Stacking for IoT (DIS-IoT) combining Multilayer Perceptron (MLP), DNN, CNN and LSTM is used in TON-IOT and CICIDS2017.

It is clear that the reviewed papers demonstrate variety of algorithms and methods, ranging from CNNs and LSTMs to more complex hybrid models. Such approaches were applied to various IoT datasets. Things that demonstrate that researchers are based different strategies to improve IoT security. Hence, this reflects the continuous development of techniques to meet the increasing challenges of protecting IoT networks.

D. Performance Analysis of the Different Models Studied

As the IoT continues to grow, ensuring security in these environments has become increasingly challenging. Many different algorithms and techniques have been developed to detect and prevent intrusions. Each method has its own strengths and weaknesses, depending on the situation. This section looks at the various studies and approaches used to address this issue, particularly focusing on network and host architectures (Table A2). The studies cover different algorithms such as CNN, LSTM networks, and hybrid models, showing how a variety of techniques are being used to improve IoT security [20].

Many studies have shown the optimal algorithms and datasets for intrusion detection tasks. For binary classification, Long Short-Term Memory-Pearson Correlation Coefficient-Extreme Gradient Boosting (LSTM-PCC-XGBoost) achieved 99.99% accuracy on the Bot-IoT dataset [21]. Similarly, Improved Hybrid Lightweight Neural Architecture (IHLNA) achieved impressive results on the NSL-KDD dataset, with a 99.96% accuracy [22]. For the Message Queuing Telemetry Transport-Internet of Things Intrusion Detection System 2020 dataset (MQTT-IoT-IDS2020) dataset (Uni-flow), an unknown algorithm got 99.70% accuracy [23]. These findings demonstrate the capability of some of the models to address the binary classification problems in various datasets.

For multiclass classification problems, CNN-LSTM has been shown to be effective with a 99.60% accuracy on the CIC-IDS2017 data as reported there [24]. On the UNSW-NB15 dataset, CNN-LSTM also performed well, achieving 94.77% accuracy [25]. The DNN-BiLSTM model worked very well on the Canadian Institute for Cybersecurity Internet of Things 2023 (CICIoT2023) dataset with 93.13% classification accuracy [26]. These case studies show that models using CNN and LSTM components are appropriate for the multiclass classification task in network intrusion detection.

Among the most popular datasets, NSL-KDD has been widely evaluated. IHLNA reached 99.96% accuracy on this dataset [22]. The model of CNN-BiLSTM-Attention achieved 91.07% accuracy [27]. The Deep Neural Decision Forest (DNDF) model, combined with Principal Component Analysis (PCA) preprocessing, obtained an accuracy of 98.38% [28]. Among the CIC-IDS2017, the DNN-BiLSTM obtained the highest (99.67% score [26]), as well as the second-highest (98.73% score [29]), respectively. UNSW-NB15 has also demonstrated potential application to LSTM-GRU achieving 99.99% [30]. And these datasets still work as benchmarks for evaluating and optimizing the models in the community.

In general, hybrids architectures such as CNN-LSTM, DNN-BiLSTM and LSTM-PCC-XGBoost are superior to pure models for addressing difficult intrusion detection problems. Refs. [21, 24, 26] demonstrate the benefits of use of these integrative methods. Further, by way of example, PCA-based pre-processing [28], and the inclusion of datasets such as Bot-IoT and MQTT-IoT-IDS2020, which are related to the Internet-of-Things, shows the direction of the field—towards solving IoT security problems. These studies also highlight the requirement of new models for effectively countering emerging threats.

This research compilation offers an overview of the latest advances in the area of intrusion detection, showcasing the different approaches and algorithms employed by researchers to enhance security in IoT environments, where security challenges are particularly complex and evolving.

E. Analysis of Intrusion Detection Systems: Architectures, Algorithms, and Methodologies

Studies reporting on IDS and architectures and algorithms based on them are reported in Table A3. All rows represent an IDS architecture implemented by a different study, including the algorithms, method of implementation, and the pros and cons of each approach.

IDS architectures and algorithms used: In this research, various IDS architecture, including that of network and host, as well as various algorithms including CNN, LSTM, Autoencoders (AE), and Deep Neural Networks (DNN) etc., are proposed.

Methodology: All the studies propose a different way for intrusion detection and usually are a mixture of models, even deep learning architectures.

Advantages: The benefit of the approaches varies, depending on the method, but some works report encouraging results, with respect to the accuracy, successful attack detection, real-time response and low computational complexity.

Disadvantages: The drawbacks listed are generally the complexity increases, attack types not detectable, model transfer limitations, dataset.

Algorithms examined in the papers employ different deep learning and hybrid methods to perform intrusion detection in IoT systems. For example, the Genetic Algorithm Feature Reduction Convolutional Neural Network (GA-FR-CNN) and DNN-CNN-LSTM-RNN models integrate Convolutional Neural Networks (CNNs) with Long Short-Term Memory (LSTM) networks, with good sensitivity and robustness in detection. Yet, such models, e.g., GA-FR-CNN [31], are computationally expensive and demand heavy resources for training and inference. In particular, the effectiveness of such methods greatly depends upon the set of data used, meaning they are far from being easily generalizable to real-world deployments.

Computational efficiency is one of the most concem of its studies. Although deep learning models such as LSTM-RNN and Random Forest (RF)-Artificial Neural Network (ANN)-LSTM-GRU have been shown to deliver high accuracies, they require extensive computing time and computing power, which may be unsuitable in environments with limited computational power, as reported in the LSTM-RNN study [32]. In contrast, models like TabNet in Ref. [33] aim to reduce the computational load while maintaining strong performance. Yet even these more efficient models have scalability limitations, when used to work on bigger or more complex datasets. In addition, the possibility of adversarial attack on these models, as evidenced in work such as DNN-Decision Tree (DT)-RF [34], is another key issue to be addressed for deploying these models in security-critical IoT platforms.

Dataset dependency and the need for extensive preprocessing are recurring issues across the reviewed models. Some models, e.g., Deep Transfer Learning (DTL) [34] and CNN-LSTM [31], achieve promising performance on limited datasets, but the performance drops whenever they are applied to another/unseen data. This illustrates the role that borrowing eclectic data sets can play in achieving generalization of intrusion detection models. On the other hand, for certain models such as AE-Reinforcement Learning (RL) [35], complex feature engineering/preprocessing is necessary, so as to increase the system complexity, which may also restrict its practical use as a real-time system in dynamic and fast-changing IoT environment.

Interpretability and transparency are often sacrificed for improved accuracy in many models. In particular, although models such as CNN-BiLSTM-Attention [27] and LSTM-Tensor Processing Unit (TPU) [36], while achieving high detection accuracy, are not transparent enough so as to be able to explain how the classifiers reason. This is especially critical in real-world applications, where the model's decision-making can be understood. Hybrid models, such as CNN-GRU in [29] and RF-ANN-LSTM-GRU [37], offer a balance between accuracy and robustness, but their complexity can pose challenges in terms of resource utilization and explainability, making them less suitable for environments with strict resource constraints.

These models are highly accurate and robust, but they often require significant hardware resources and continuous evaluation of their performance in real time.

In these various studies on intrusion detection in IoT using deep learning, the limits of computational capabilities are critical. IoT devices have restricted computing and storage resources, along with bandwidth constraints. To adapt, deep learning models must be tailored by simplifying algorithms and reducing complexity to maintain adequate performance. The storage and processing of data on IoT devices are also challenging due to storage and bandwidth restrictions. Therefore, models must be adjusted to handle smaller datasets or preprocess data locally before sending it for further analysis. In summary, the challenge for researchers has been and will continue to be the creation of IDS that protect data while maintaining high efficiency.

F. Computational and Resource Constraints

The usage of deep learning faces several obstacles. Accessing representative IoT datasets is difficult, which complicates the generalization of results. Moreover, the complexity of deep learning models sometimes makes it challenging to understand the decisions made by the system. Limitations in computing power and memory on IoT devices restrict the feasibility of certain models. Finally, the security of IoT data is a major concern, with risks of adversarial attacks aimed at manipulating data or circumventing the detection system.

V. CONCLUSION

This study underscores the critical importance of advancing Intrusion Detection Systems (IDSs) to enhance the security of IoT networks, particularly in the face of escalating threats such as denial-of-service attacks and data breaches. Recent breakthroughs in artificial intelligence, notably deep learning, have shown immense potential to improve the detection capabilities and overall performance of IDSs by accurately identifying malicious activities amidst increasing network traffic. Our review focused on key features related to IDS detection and the application of deep learning algorithms. Hence, we primarily highlight the prevalent use of CNN and LSTM models in recent research. Then we show that the combination of these approaches has demonstrated significant improvements in accuracy and precision.

Our Future work should prioritize the development of novel IDS architectures tailored to the unique challenges of IoT environments, such as low computational resources, real-time detection requirements, and evolving attack vectors. Specifically, exploring hybrid models that integrate various deep learning techniques, incorporating explainability for better interpretability, and designing lightweight algorithms suitable for deployment on resource-constrained IoT devices are promising directions. Additionally, there is a need to build comprehensive and diverse datasets for training and testing to ensure robustness against new and sophisticated threats. Overall, continuous innovation in IDS design remains vital to safeguarding IoT networks in an increasingly connected world.

APPENDIX A: SUPPLEMENTARY TABLES FOR INTRUSION DETECTION RESEARCH

TABLE A1. DATASETS USED FOR INTRUSION DETECTION

Dataset	Year of publication	Features	Number of classes	Total normal records	Total attack records
Telemetry and Operational Networks-Internet of Things dataset (TON-IOT)	2020	44	-	-	161043
NSL-KDD	1998	41	5	77054	71463
UNSW-NB15	2015	49	10	2218761	321283
BoT-IoT	2018	46	5	477	3668045
CICIDS 2017	2018	77	7	2273097	557646
CSE-CIC-IDS2018	2018	75	-	2856035	1669364
Washington University in St. Louis-Industrial Internet of Things 2021 dataset (WUSTL-IIoT-2021)	2021	-	5	1106747	87014
Knowledge Discovery and Data Mining Cup 1999 dataset (KDDCup-99)	1998	41	23	-	-
Edge-based Industrial Internet of Things dataset (Edge-IIoT)	2022	-	15	1091198	9728708
Wireless Sensor Network Dataset (WSN-DS)	2016	23	5	340066	34595
Washington University in St. Louis-Embedded Healthcare Monitoring System 2020 dataset (WUSTL-EHMS-2020)	2020	44	-	14272	2046
MQTT-IoT-IDS2020	2020	83	5	-	-
Botnet Internet of Things-Lab 01 dataset (BoTNeTIoT-L01)	2019	34	2	-	-
Network Flows-Botnet Internet of Things dataset (NF-BoT-IoT)	2020	12	5	13859	586241
Network Flows-Telemetry and Operational Networks Internet of Things dataset (NF-ToN-IoT)	2020	12	10	270279	1108995
NF-CSE-CIC-IDS2018	2020	12	7	7373198	1019203
Network Flows-University of New South Wales Network-Based 2015 dataset (NF-UNSW-NB15)	2020	12	9	1550712	72406
Network Flows-University of Queensland Network Intrusion Detection System dataset (NF-UQ-NIDS)	2020	12	21	9208048	2786845
Intrusion Detection in Software-Defined Networks dataset (InSDN)	2020	-	8	343939	275515
CICIDS 2023	2023	46	8	1098195	45588384
IoT Intrusion	2019	83	4	40073	5805710
Distributed Smart Software-Defined Operating Systems dataset (DS2OS)	2018	13	8	357962	10027
Aegean Wi-Fi Intrusion Dataset (AWID)	2015	154	4	2163975	207243
IoT-23	2020	21	10	-	-
CIC-DDoS2019	2019	88	2	50000	12000000
Extended Industrial Internet of Things Intrusion Detection dataset (X-IIoTID)	2021	68	-	421417	399417

TABLE A2. PERFORMANCE OF THE DIFFERENT MODELS STUDIED

Paper	Algorithm	Dataset	Accuracy	Precision	Recall	F1-Score
[21]	LSTM-PCC-XGBoost	Bot-IoT (LSTM) (binary classification)	99.99%	99.99%	100%	99.99%
		EdgeIoT (PCC-LSTM) (binary classification)	99.99%	99.99%	100%	98.26%
[22]	IHLNA	KDD-CUP99	99.56%	99.67%	-	-
		NSLKDD	99.96%	99.67%	-	-
		UNSW-NB-15	99.96%	99.67%	-	-
[23]	Algorithm not mentioned	MQTT-IoT-IDS2020 (Bi-flow)	99.56%	99.60%	99.60%	99.60%
		MQTT-IoT-IDS2020 (Uni-flow)	99.67%	99.70%	99.70%	99.70%
[24]	ANN-CNN-LSTM	IoT-23 (.DDOS)	99.6%	93.5%	99.1%	96.3%
[25]	CNN-LSTM	UNSW_NB15	96.08%	96.08%	96.08%	96.08%
	ANN		97.01%	97.01%	97.01%	97.01%
[26]	DNN-BiLSTM	CIC IDS2017	99.67%	99.54%	99.67%	99.59%
		N-BaIoT	99.98%	99.98%	99.98%	99.98%
		CICIoT2023	93.13%	91.80%	93.13%	91.94%
[27]	CNN-BiLSTM-Attention	NSL-KDD	90.01%	90.35%	91.07%	90.71%
[28]	DNDF (CNN variant)	NSL-KDD (PCA)	98.38	98.08%	98.69%	98.38%
		CICIDS2017 (PCA)	98.84%	99.12%	98.56%	98.84%
		UNSW-NB15 (PCA)	98.23%	96.90%	99.65%	98.25%
[29]	CNN-GRU	CIC-IDS2017	98.73%	-	-	-
[30]	LSTM-GRU	IoT23	98.12%	98.06%	98.31%	98.18%
		UNSW-NB15	99.98%	99.99%	99.98%	99.99%

[31]	FR-CNN; GA-FR-CNN	UNSW-NB 15 (AAFSA with GA-FR-CNN)		94.48%	94.29%	94.56%	94.42%
		BoT NeT IoT (AAFSA with GA-FR-CNN)		93.77%	86.66%	95.87%	91.03%
[32]	LSTM	Software-Defined Networks Internet of Things dataset (SDN-IoT), Software-Defined Networks Network Flows – TJ dataset (SDN-NF-TJ)		97.10%	-	-	-
[33]	Tabular Neural Network (TabNet)	CIC-IDS2017		97%	-	-	-
		CSE-CICIDS2018		95%	-	-	-
		CIC-DDoS2019		98%	-	-	-
[34]	DTL	IoT Intrusion (Target: Bot-IoT)		99.94%	99.94%	100%	99.97%
		Bot-IoT (Target: IoT Intrusion)		99.94%	99.94%	100%	99.97%
[35]	AE-Reinforcement Learning (RL)	BOT-IOT (Class:Normal)		99.98%	42.46%	59.61%	100%
		BOT-IOT (Class:DOS)		94.59%	96.43%	94.76%	93.14%
[36]	LSTM-TPU	BoT-IoT		99.94%	99.94%	99.94%	-
		Edge-IIoT		99.99%	100%	100%	-
		NSL-KDD		99.71%	99.70%	99.71%	-
[37]	RF-LSTM-GRU- ANN	Extended Industrial Internet of Things Intrusion Detection dataset (X-IIoTID)		99.72%	-	-	-
[38]	Fully Connected layer (FC)	Average of five attacks		93.74%	93.71%	93.82%	93.47%
[39]	CNN	CSE-CIC-IDS2018		99.65%	99.16%	98.70%	99.09%
[40]	CNN-LSTM	CSE-CIC-IDS2018, MQTT-IoT-IDS2020, BoTNeTIoT-L01		99.86%	99.80%	99.87%	99.83%
[41]	Dugat-LSTM	TON-IOT, NSL-KDD		98.76%	96.98%	97.87%	97.23%
[42]	AE, Mutual Information (MI), Genetic Algorithm (GA), LSTM	BoT-IoT		99.94%	99.94%	99.94%	-
		Edge-IIoT		99.99%	100.00%	100.00%	-
		NSL-KDD		99.71%	1.84%	99.71%	-
[43]	DNN	N-Botnet Internet of Things dataset (N-BaIoT)		97.21%	91.41%	87.31%	88.48%
[44]	CNN+DNN+RNN	CICDIoT2023 - CNN3		96.37%	96.15%	96.37%	95.51%
		CICDIoT2023 - DNN3		88,64%	91.20%	88.64%	88.51%
		CICDIoT2023 - RNN1		96.52%	96.25%	96.52%	95.73%
[45]	CNN+DNN	UNSW-NBnew		81.00%	-	-	-
		KDD-CUP		99.20%	-	-	-
		UNSW-NB		81.00%	-	-	-
[46]	Elman Recurrent Neural Network (ERNN)	-		98.52%	96.00%	98.00%	-
[47]	CNN-LSTM hybride	IoT-23		95.00%	-	-	-
		CICID2017		98.99%	-	-	-
		N-BaIoT		99.99%	-	-	-
[48]	CNN	NF-bot-IoT			100.00%	80.00%	89.00%
[49]	Feedforward Neural Network (FNN)-Focal + CNN-Focal	Bot-IoT	FNN-Focal	91.55%	55.59%	63.80%	57.84%
			CNN-Focal	86.77%	61.65%	63.25%	58.53%
		WUSTL-EHMS-2020	FNN-Focal	93.26%	95.24%	73.69%	80.11%
			CNN-Focal	93.08%	94.23%	73.38%	79.63%
		WUSTL-IIoT-2021	FNN-Focal	98.95%	77.22%	64.06%	68.48%
			CNN-Focal	98.21%	88.54%	66.51%	70.50%
[50]	AE	Traffic from different IoT+ Bot-IoT devices		-	99.99%–100%	99.94%–99.97%	99.96%–99.98%
[51]	RNN-GRU	ToN-IoT		99.00%	99.00%	98.00%	97.00%
[52]	Deep Intrusion System for IoT (DIS-IoT) (MLP, DNN, CNN, LSTM)	Binary classification	ToN_IoT	99.60%	99.40%	99.40%	99.40%
			CICIDS2017	98.70%	95.90%	97.60%	96.70%
			Secure Water Treatment dataset (SWaT)	99.60%	99.70%	99.90%	99.80%
[53]	LSTM	Binary classification	NSL-KDD	81.10%	92.10%	73.20%	81.50%
			UNSW-NB15	86.60%	81.10%	98.80%	89.10%
			ToN_IoT	87.30%	78.40%	88.00%	82.90%
[54]	FFNN, LSTM, Random Neural Network (RandNN)	CIC-IoT22	FFNN	99.84%	99.93%	99.93%	99.93%
			LSTM	99.78%	99.89%	99.89%	99.89%
			RandNN	96.42%	96.42%	96.42%	96.42%
[55]	Singular Value Decomposition (SVD)+ (LSTM,Bagging tree,Bi-LSTM, K-Nearest Neighbors (KNN),GRU)	Binary classification		99.99%	-	-	-
		Multiclass classification		99.98%	-	-	-
[56]	CNN-LSTM, CNN-GRU	CNN-LSTM		99.73%	99.70%	99.90%	99.80%
		CNN-GRU		99.60%	99.50%	99.90%	99.70%
[57]	Enhanced Intrusion Detection Model (EIDM) (MLP, CNN, LSTM, CNN+LSTM)	CICIDS2017	Multiclass classification	95.00%	-	-	-
[58]	Neighborhood Search Binary Particle Swarm Optimization – Deep Convolutional Neural Network (NSBPSO-DCNN)	-		98.86%	99.03%	-	-
[59]	AE	-		99.65%	99.99%	99.85%	99.55%

[60]	Densely Connected Convolutional Network (DenseNet) and inception time(CNN)	ToN-IoT	Inception time	97.70%–99.90%	95.97%–99.90%	95.91%–99.90%	95.91%–99.90%
		Edge-IIoT	DenseNet	94.94%	98.30%	92.40%	95.30%
		UNSW-NB15	DenseNet	98.60%	98.90%	98.40%	98.70%
[61]	Generative Adversarial Network (GAN)-DNN	UNSW-NB15	DNN	84.00%	-	-	-
			GAN-DNN	91.00%	-	-	-
[62]	CNN	NID		99.51%	-	-	-
		Bot-IoT		95.55%	-	-	-
[63]	CNN-LSTM	CIC-IDS2017	Binary classification	99.64%	-	-	99.56%
			Multiclass classification	99.60%	-	-	99.60%
		UNSW-NB15	Binary classification	94.53%	-	-	94.69%
			Multiclass classification	82.41%	-	-	94.77%
		WSN-DS	Binary classification	99.67%	-	-	98.00%
			Multiclass classification	98.83%	-	-	98.44%
[64]	CNN,LSTM,GRU	Bot-IoT	CNN	99.70%	99.60%	99.90%	99.80%
			LSTM	99.80%	99.70%	100.00%	99.80%
			GRU	99.60%	99.60%	100.00%	99.80%
[65]	DNN, LSTM, CNN	CIC-IDS2017	DNN	94.61%	80.85%	84.60%	84.60%
			LSTM	97.67%	94.96%	95.00%	93.55%
			CNN	98.61%	97.05%	96.95%	98.09%
[66]	Recurrent Long Short-Term Memory (RLSTM)	NSL-KDD	Average (DOS attack, Normal)	98.60%	98.60%	98.60%	98.60%
		CICIDS-2017	Average (DOS attack, Normal)	99.20%	99.23%	99.22%	99.22%
[67]	-	NetFlow (NF-BoT-IoT, NF-ToN-IoT, NF-CSE-CIC-IDS2018, NF-UNSW-NB15, NF-UQ-NIDS)		93.02%	-	-	-
[68]	Temporal Convolutional Network (TCN); AE-TCN; AE-LSTM; AE- Bidirectional Recurrent Neural Network (BRNN); AE- Bidirectional LSTM (BLSTM); CNN-LSTM	Bot-IoT		100%	-	-	-
		CICIDS2017		99.90%	-	-	-
		NSL-KDD		79.00%	-	-	-
		UNSW_NB15		97.90%	-	-	-
		N-BaIoT		90.90%	-	-	-
		KDD CUP 99		99.90%	-	-	-
[69]	GAN	Aggregated Dataset		98%	98%	-	-
[70]	LSTM	ToN-IoT		97.50%	98.40%	97.90%	98.05%
		Intrusion Detection in Software-Defined Networks dataset (InSDN)		99%	-	99.60%	99.3%
[71]	DNN-Deep Belief Network (DBN)	Vehicle network packets		91.26%	-	90.96%	-
[72]	CNN-Bi-LSTM	UNSW-NB15		83.18%	83.18%	83.70%	81.19%
[73]	RNN-GRU	ToN-IoT (Multiclass classification)		88%	86%	97%	88%
		ToN-IoT (Binary classification)		99%	98%	99%	98%
[74]	Dynamic Parameter Attention (DPA)- Local Spatial Convolutional Neural Network (LSCNN)	NSL-KDD		91.70%	81.50%	79.88%	80.68%
[75]	CNN	MQTTIOT-IDS2020		99.74%	-	-	-
[76]	DNN-CNN-LSTM-RNN	CICIoT2023 (Binary)		99.76%	-	-	-
		CICIoT2023 (Multiclass)		91.27%	-	-	-
[77]	Sine–Cosine Harmonic Oscillator (SHO)-LSTM	Benchmark datasets		99.89%	98%	97.5%	99%
[78]	Deep convolution network (DCN)	NSL-KDD		97.29%	-	-	-
[79]	DNN	Distributed Smart Software-Defined Operating Systems dataset (DS2OS) (multi-class)		99.41%	99%	99%	99%
	DT			99.44%	99%	99%	99%
	RF			99.44%	99%	99%	99%
[80]	CNN- Stacked Autoencoder (SAE)	AWID (Two classes)		99.77%	97.95%	99.09%	98.51%
[81]	LSTM-RNN; DNN; Residual Network (ResNet)	N-BaIoT		99.8%	-	-	-
		UNSW-NB15		99.82%	-	-	-
[82]	SAE-CNN	Bot-IoT		99.9%	99.9%	100%	99.9%
[83]	CNN	IoT-23 (Meta-learner: RF)		99.90%	99.83%	99.97%	99.90%

TABLE A3. DETAILED SUMMARY OF ALGORITHMS, METHODOLOGIES, ADVANTAGES, AND LIMITATIONS FROM REVIEWED PAPERS

Paper	Algorithm Used	Dataset Used	Language Used	Methodology	Advantages	Limits
[21]	LSTM-PCC-XGBoost	BoT-IoT; Edge-IIoT	Python	Feature selection; Outlier detection; Classification; Evaluation; Optimization	Improved performance on imbalanced datasets; Robust metrics; Scalability potential; Future enhancements	Computational complexity; Edge computing optimization needed; Scalability challenges
[22]	IHLNA	KDD-CUP99; NSLKDD; UNSW-NB-15	MATLAB	Data preprocessing and feature selection; Data up-sampling; Layered network model; Model training	High accuracy and performance; Low false acceptance rate; Cost-effective	Dataset dependency; Limited real-world validation; Potential overfitting
[23]	Algorithm not mentioned	MQTT-IoT-IDS2020	WEKA	Data collection; Classification; Performance evaluation	High detection accuracy; Focus on MQTT protocol; Scalability	Binary classification limitation; Dataset-specific optimization; Interpretability
[24]	ANN-CNN-LSTM	IoT-23	Python	Data preprocessing; Model design and training; Real-time analysis; Evaluation and comparison	Scalability; Real-time detection; Flexibility	Complexity; Resource intensive; Dataset dependency; Lack of real-world validation
[25]	CNN-LSTM; ANN	UNSW_NB15	Python	Model 1: CNN-LSTM; Model 2: ANN with fully connected layers; Training and evaluation	High performance; Hybrid approach; Custom architecture	Dataset dependency; Computational cost; Model complexity
[26]	DNN-BiLSTM	CIC IDS2017; N-BaIoT; CICIoT2023	Python	Model architecture; Feature dimensionality reduction; Dynamic quantization; Training and evaluation	Enhanced detection accuracy	Implementation complexity; Dataset specificity
[27]	CNN-BiLSTM-Attention	NSL-KDD	Python	Feature extraction; Feature fusion and alignment; Model design; Evaluation	High accuracy; Effective detection; Model interpretability	High computational complexity and resource requirements; dataset dependency
[28]	DNDF	NSL-KDD; CICIDS2017; UNSW-NB15	Python	Feature selection; Model development; Performance optimization; Comparative analysis	High accuracy; Efficiency with limited features; Fast prediction; Versatility; Feature selection integration	Dependency on feature selection; Complexity; Dataset variability; Scalability challenges
[29]	CNN-GRU	CIC-IDS2017	Python	Dataset preprocessing; Feature learning; Model training; Comparative analysis	High accuracy; Feature learning; Reduced false alarms; Versatility	Dataset dependency; Computational overhead; Explainability; Imbalanced dataset
[30]	LSTM-GRU	IoT-23; CICIDS2017	Python	Hybrid metaheuristics-deep learning approach; Feature selection; RNN models	Improved intrusion detection; Optimized feature selection; Handling diverse IoT attacks	Complexity; Dependence on feature selection; Scalability
[31]	FR-CNN; GA-FR-CNN	UNSW-NB 15; BoT NeT IoT	MATLAB	Feature selection with AAFSO; Model training with GA-FR-CNN; Dataset evaluation	High accuracy; Optimized feature selection; Generalizability	Complexity of GA-FR-CNN; Dataset dependency; Training time
[32]	LSTM	SDN-IoT; SDN-NF-TJ	Python	Dataset preprocessing; Model design; Validation; Performance analysis	Outperforms traditional ML and other DL models in classifying attacks; High generalizability	vulnerable to adversarial attacks; Higher computational resource
[33]	TabNet	CIC-IDS2017; CSE-CICIDS2018; CIC-DDoS2019	Python	Data preprocessing; Feature selection; Model training; Evaluation	High accuracy; Interpretability; Tabular data efficiency	Resource intensive; Dataset dependency; Limited real-world validation; Complexity
[34]	DTL	IoT Intrusion; Bot-IoT	Python	Transfer learning framework; Dataset selection process; Pre-training using transfer learning; Implementation and evaluation	Efficient selection of source domain; Universal applicability; Extensibility	Dependence on dataset selection; Time-to-accuracy optimization; Lack of detail on neural network architecture
[35]	AE-RL	BOT-IOT	Not mentioned	Simultaneous fine-tuning of the environment; Classifier embedded in the RL model; Evaluation using Bot-IoT dataset	Improved performance; Adversarial strategy; Innovative framework	Computational complexity; Dataset dependency; Potential overfitting

[36]	LSTM-TPU	Enhanced BoT-IoT; Edge-IIoT; NSL-KDD	Python	Model development; TPU utilization; Dataset testing; Performance comparison	High accuracy; Fast processing; Robust evaluation; Integration with IoT standards; Scalability	Dependence on TPUs; Feature engineering complexity; Limited metric explanation; Dataset bias; Real-world challenges
[37]	RF-ANN-LSTM-GRU	X-IIoTID	Python	Intrusion detection; Data security via blockchain; Smart contract classification	High accuracy; Integration of AI and blockchain; Comprehensive security layers; Prevention mechanism	Computational complexity; Scalability challenges; Adversarial vulnerability
[38]	FC	Model trained with the dataset produced by the experimental system	Python	Developed a four-layer deep fully connected neural network; Attack detection; Experimental validation	High accuracy; Protocol independence; Broad attack coverage; Real-time detection	Dataset dependence; Computational complexity; Limited attack types; Potential false positives
[39]	CNN	CSE-CIC-IDS2018	Python	Dataset preprocessing; Model architecture (Five convolutional layers); Experimental validation	High performance; Automated feature extraction; Wide applicability	Dataset dependency; Potential overfitting; Limited interpretability
[40]	CNN-LSTM	CIC-IDS2018, MQTT-IoT-IDS2020, BoTNeTIoT-L01	Python	Data preprocessing; Model training and testing; Assessment	High accuracy; Dual detection strategies; Feature reduction; Generalization	Computational cost; Dataset dependency; Limited real-time validation; Scalability concerns
[41]	Dugat-LSTM	TON-IOT, NSL-KDD	Not mentioned	Preprocessing; Balancing; Feature handling; Model training; Evaluation	High accuracy; Class imbalance handling; Feature optimization; Robust model	Complexity; Computational cost; Generalization challenges; Dependence on preprocessing
[42]	AE; MI; GA; LSTM	BoT-IoT; Edge-IIoT; NSL-KDD	Python	Dataset preparation; Feature engineering; Model architecture; Implementation; Evaluation	High accuracy; Low latency; Adaptability; Temporal analysis	Resource dependency; Complexity; Scalability challenges; Dataset-specific performance
[43]	DNN	N-BaIoT	Python	Dataset preparation; Model training; Ensemble averaging; Validation	High accuracy; Adaptability; Improved generalization; Scalability	Computational overhead; Latency; Dataset dependency; Complexity
[44]	CNN+DNN+RNN	CICDIoT2023	Not mentioned	Dataset preparation; Model Implementation; Evaluation	High detection accuracy; Temporal pattern recognition; Applicability to realistic scenarios	Computational intensity; Potential overfitting; Limited generalization
[45]	CNN+DNN	UNSW-NBnew; KDD-CUP; UNSW-NB	Not mentioned	Dataset Preparation; Model development; Explainability	High classification accuracy; Feature reduction; Explainability	Computational complexity; Generalization; Interpretability limitations
[46]	ERNN	KDDCup-99; NSL-KDD	MATLAB	Dataset preparation; Feature selection; Model development; Performance evaluation	High accuracy; Effective feature selection; Robust optimization	Dataset limitations; Computational complexity; Generalization
[47]	Hybrid CNN-LSTM	IoT-23; CICID2017; N-BaIoT	Not mentioned	Dataset preparation; Feature extraction and modeling; Model optimization; Validation and testing	High accuracy; Adaptability; Efficient deployment	Model complexity; Limited real-world validation; Dependence on PCA
[48]	CNN	NF-bot-IoT	Python	Data collection; Feature extraction with CNN; Classification with XGBoost; Model training and testing	High detection accuracy; Effective feature extraction; Scalability	Computational power; Data dependency; Model complexity
[49]	FNN-Focal + CNN-Focal	Bot-IoT; WUSTL-EHMS-2020; WUSTL-IIoT-2021	Not mentioned	Data collection and preprocessing; Handling data Imbalance with focal loss; Training and evaluation; Comparison with state-of-the-art approaches	Improved performance on imbalanced data; Better generalization; State-of-the-art comparison	Computational complexity; Dataset dependency; Model interpretability
[50]	AE	Traffic from different IoT+ Bot-IoT devices	Python	Data preparation; Model architecture; Detection; Evaluation	Device independence; Lightweight; Transferability; High accuracy	Semi-supervised limitation; Unseen anomalies; False positive sensitivity

[51]	RNN-GRU	ToN-IoT	Python	Three-layered IoT system; The model consists of RNN-GRU networks; Training and Testing; Optimization; Comparison with Other Techniques	Cross-Layer Attack detection; High accuracy; Adaptability to new attacks; Optimization efficiency	Resource intensity; Sensitivity to dataset quality; Complexity of hyperparameter tuning
[52]	MLP, DNN, CNN, and LSTM	ToN_IoT; CICIDS2017; SWaT	Python	Stacking ensemble approach; Model training; Comparison with other models	Improved performance with ensemble learning; Effective multi-class classification; Low false positive rate; Scalability	Complexity; Resource consumption; Dependence on high-quality data
[53]	LSTM	NSL-KDD, UNSW-NB15, ToN_IoT	Python	Model training; Explainability via SPIP; Evaluation	High detection accuracy; Interpretability; Real-time capability; Generalization	Complexity of model; Dependence on quality data; Interpretability overhead
[54]	FFNN, LSTM, RandNN	CIC-IoT22	Not mentioned	Model development; Training and testing; Comparison	High accuracy; Adaptability; Fast response time; Wide applicability	Computational complexity; Overfitting risk; Model complexity
[55]	SVD+(LSTM, Bagging tree, Bi-LSTM, KNN, GRU)	TON-IOT	Not mentioned	Data preprocessing; Model training; Evaluation	High accuracy; Addressing class Imbalance; Feature reduction for efficiency; Versatility for binary and multi-class classification	Dependency on the ToN_IoT dataset; Computational complexity; Overfitting risk; Scalability in large-scale deployments
[56]	CNN-LSTM, CNN-GRU	NSL-KDD	Not mentioned	Data preprocessing; Model training; Model evaluation	High performance; Effective at capturing sequential patterns; Hierarchical feature learning; Robustness to variability	High computational cost; Complexity; Data requirements; Risk of overfitting; Interpretability
[57]	EIDM (MLP, CNN, LSTM, CNN+LSTM)	CICIDS 2017	Python	Data preprocessing; Model training; Model evaluation	High accuracy; Multi-class classification; Deep learning approach; Comparison with other models	Computational complexity; Data dependency; Overfitting risk; Interpretability; Time complexity for real-time detection
[58]	NSBPSO-DCNN	BoT-IoT, UNSW-NB15	MATLAB	Algorithm design; Data preprocessing; Model training; Model evaluation	Improved optimization; Higher detection accuracy; Adaptability; Hybrid optimization	Computational complexity; Dependence on data quality; Overfitting risk; Real-time processing
[59]	AE	KDDCup-99	Python	Model design; Training process; Validation; Implementation	Offers high accuracy, precision, recall, and F1-score with reduced training time	Computational complexity; The results obtained are less stable due to the reduced number of training data
[60]	DenseNet and Inception Time (CNN)	ToN-IoT, Edge-IIoT, UNSW-NB15	Not mentioned	Data preprocessing; Model training; Evaluation and comparison; Sliding window approach (Inception Time)	High accuracy; Versatility; Time-series data handling; Effective multi-class classification	Computational cost; Overfitting risk; Data quality and representation; Interpretability
[61]	GAN-DNN	UNSW-NB15	Python	Data preprocessing; Model training; Class imbalance solution; Evaluation	High accuracy after balancing; Improved performance with feature selection; Handling class imbalance	Computational cost; Overfitting risk with GANs; Data quality and representativeness; Interpretability of the DNN model
[62]	CNN	Bot-IoT	Python	Data preprocessing; Model architecture; Training and testing	High accuracy; Scalability; Adaptability to IoT traffic; Ability to detect complex attacks	Computational complexity; Overfitting risk; Interpretability; Data dependency
[63]	CNN-LSTM	CIC-IDS2017, UNSW-NB15, WSN-DS	Python	Data preprocessing; Model architecture; Model training; Model evaluation	High detection rate; Robust performance; Automated feature extraction; Reduced false alarm rate	Computationally expensive; Data dependency; Complexity of model tuning; Risk of overfitting
[64]	CNN, LSTM, GRU	Bot-IoT	Python	Data preprocessing; Model training; Model evaluation	Reproducible dataset; High accuracy; low false alarms; SOTA performance benchmarking	Generalization limited; Few features; Absent computational & real-time analysis

[65]	DNN	CIC-IDS2017	Not mentioned	Model development; Comparison	High accuracy; Scalability; Flexibility; Feature learning	Computational cost; Overfitting risk; Data dependency; Interpretability
[66]	RLSTM	NSL-KDD	MATLAB	Data preprocessing; Model training; Performance comparison	High performance; Temporal dependency learning; Reduced human intervention; Scalability	Computational cost; Data dependency; Overfitting risk; Interpretability
[67]	DNN	NF-UQ-NIDS, NF-UNSW-NB15, NF-CSE-CIC-IDS2018	Python	Packet capturing and detection; Dataset preparation; Model training and evaluation	Real-time detection; High accuracy; Suitability for IoT constraints; Automatic feature extraction	High computational requirements; Potential for overfitting; Limited interpretability; Dependency on dataset quality
[68]	TCN; AE-TCN; AE-LSTM; AE-BRNN; AE-BLSTM; CNN-LSTM	Bot-IoT; CICIDS2017; NSL-KDD; UNSW_NB15; N-BaIoT; KDD CUP 99	Not mentioned	Dataset analysis; Classifier evaluation; Comparison and benchmarking; Empirical results	Comprehensive evaluation; Reduced bias; Improved IDS design	Computationally intensive; Dependency on dataset quality; Complexity
[69]	GAN	Aggregated Dataset	Python	Problem addressing; Model development; Preprocessing; Evaluation	Improved detection; Reduced false positives; Scalability; Data augmentation	Computational complexity; Threshold sensitivity; Model interpretability; IoT-specific scope
[70]	LSTM	ToN-IoT; InSDN	MATLAB	Feature selection; Model development; Performance analysis; Routing protocol integration	High performance; Feature selection; IoT-specific focus; Integration with routing protocols	Computational requirements; Complexity; Dataset dependency; Scalability
[71]	DNN-DBN	Vehicle network packets	MATLAB	Model training; Integration with DBN; Decision reporting; Real-time evaluation	High detection rate; Hierarchical clustering; Real-time capabilities; Comprehensive analysis; Decision-making support	Computational requirements; False positives; Dataset dependency; Limited generalization; Cluster head vulnerability
[72]	CNN-Bi-LSTM	UNSW-NB15	Python	Data preprocessing; Model design; Model training and testing; Comparative analysis	Higher accuracy; Improved precision; Low false positive rate; Comprehensive feature Learning; State-of-the-art performance	Complexity; Training time; Data dependency; Overfitting risk; Interpretability
[73]	RNN-GRU	ToN-IoT	Python	Data collection; Model design; Training and testing; Comparison with Other techniques	Enhanced performance; Improved data processing; Real-world applicability	Dataset dependency; High computational requirements; Limited real-world testing
[74]	DPA-LSCNN	NSL-KDD	Not mentioned	Data purification (DPA); Conversion of data to image data; Separable convolutions; LSCNN model	Improved accuracy; Reduced computational cost; Enhanced feature extraction	Increased model complexity; Training time limitations; Attribute correlation issue
[75]	CNN	MQTTIOT-IDS2020	Python	Device-specific optimizations; Model deployment; Centralized IDS in fog or cloud layers; Evaluation on multiple devices	High accuracy; Real-time Detection; Lightweight optimization	Device dependency; Limited dataset information
[76]	DNN-CNN-LSTM-RNN	CICIoT2023	Python	Preprocessing operations; Model training	Improved detection efficiency; Focused attack identification	Moderate accuracy; Scalability concerns; Vulnerability to adversarial attacks
[77]	SHO-LSTM	Aggregated Dataset	Python	Hybrid deep learning model; Real-time experimentation; Preprocessing and optimization; Testing on multiple datasets	High detection accuracy; Comprehensive performance; Scalability	Performance drop in real-time scenarios; Energy consumption considerations
[78]	DCN	NSL-KDD	Not mentioned	DCN IDS; Deep learning for IDS; Multicloud IoT integration; Optimization of training	Better overall performance; Multi-level feature extraction; Reduced training time	Computational complexity; Dataset dependence; Limited generalization

[79]	DNN-DT-RF	DS2OS	Python	Model training; Adversarial sample generation and retraining	Enhanced resilience; Generalization; Practical application	Sensitivity to adversarial attacks; Complexity; Feature engineering challenges
[80]	CNN-SAE	AWID	Not mentioned	Efficient data processing; Model design; Training and evaluation	High accuracy; Lightweight architecture; Fast response time	Dataset dependency; Limited attack types; Lack of robustness testing
[81]	LSTM-RNN; DNN; ResNet	N-BaIoT; UNSW-NB15	Not mentioned	Data preprocessing; Ensemble strategy; Hyperparameter tuning; Comparison with other models	Improved performance; Adaptability; Diversity of classifiers; Potential for zero-day attack detection	Complexity; High computational resources; Risk of overfitting; Interpretability
[82]	SAE-CNN	Bot-IoT	Python	Data preprocessing; Feature extraction; Model design and training; Training and testing; Performance comparison	High accuracy; Dimensionality reduction; Real-time capability; Low false positives	Limited dataset scope; Binary classification; Resource intensive; Scalability concerns
[83]	CNN	IoT-23	Python	Three one-dimensional convolutional neural networks are employed; Ensemble learning; Hyperparameter optimization	High accuracy; Ensemble learning; Reduced processing time; IoT-specific design	Model complexity; Dataset dependency; Training time

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

AB and MA conducted the research; SD and AG analyzed the data; AB wrote the paper; AA Admisitred the project; AG and FA corrected and revised the paper; all authors had approved the final version.

References

[1] J. S. Yalli, M. H. Hasan, and A. A. Badawi, "Internet of Things (IoT): Origins, embedded technologies, smart applications, and its growth in the last decade," *IEEE Access*, vol. 12, pp. 91357–91382, 2024.

[2] M. Zipperle *et al.*, "Provenance-based intrusion detection systems: A survey," *ACM Computing Surveys*, vol. 55, no. 7, 135, 2022.

[3] A. Thakkar and R. Lohiya, "A review on challenges and future research directions for machine learning-based intrusion detection system," *Archives of Computational Methods in Engineering*, vol. 30, pp. 4245–4269, 2023.

[4] Z. T. Sworna, Z. Mousavi, and M. A. Babar, "NLP methods in host-based intrusion detection systems: A systematic review and future directions," *Journal of Network and Computer Applications*, vol. 220, 103761, 2023.

[5] N. Odyuo, S. Lodh, and S. Walling, "Multifactor mutual authentication of IoT devices and server," in *Proc. 2023 5th International Conf. on Smart Systems and Inventive Technology (ICSSIT)*, 2023, pp. 391–396.

[6] B. Isong, O. Kgote, and A. Abu-Mahfouz, "Insights into modern intrusion detection strategies for internet of things ecosystems," *Electronics*, vol. 13, no 12, 2370, 2024.

[7] H. Liao *et al.*, "A survey of deep learning technologies for intrusion detection in internet of things," *IEEE Access*, vol. 12, pp. 4745–4761, 2024.

[8] L. Aversano *et al.*, "A systematic review on deep learning approaches for IoT security," *Computer Science Review*, vol. 40, 100389, May 2021.

[9] B. Zhang *et al.*, "Secure device-to-device communication in IoT: Fuzzy identity from wireless channel state information for identity-based encryption," *Electronics*, vol. 13, no. 5, 984, 2024.

[10] J. H. Kalwar and S. Bhatti, "Deep learning approaches for network traffic classification in the internet of things (IoT): A survey," arXiv preprint, arXiv: 2402.00920, 2024.

[11] A. Aldhaheri *et al.*, "Deep learning for cyber threat detection in IoT networks: A review," *Internet of Things and Cyber-Physical Systems*, vol. 4, pp. 110–128, 2024.

[12] S. H. Rafique *et al.*, "Machine learning and deep learning techniques for internet of things network anomaly detection—Current research trends," *Sensors*, vol. 24, no. 6, 1968, 2024.

[13] L. Thomas and S. Bhat, "Machine learning and deep learning techniques for IoT-based intrusion detection systems: A literature review," *International Journal of Management, Technology and Social Sciences (IJMTS)*, vol. 6, no. 2, pp. 296–314, 2021.

[14] S. Sharma, V. Kumar, and K. Dutta, "Multi-objective optimization algorithms for intrusion detection in IoT networks: A systematic review," *Internet of Things and Cyber-Physical Systems*, vol. 4, pp. 258–267, 2024.

[15] A. Ghaffari *et al.*, "Securing internet of things using machine and deep learning methods: A survey," *Cluster Comput.*, vol. 27, pp. 9065–9089, 2024.

[16] W. G. Hatcher and W. Yu, "A survey of deep learning: Platforms, applications and emerging research trends," *IEEE Access*, vol. 6, pp. 24411–24432, 2018.

[17] S. Muneer *et al.*, "A critical review of artificial intelligence based approaches in intrusion detection: A comprehensive analysis," *Journal of Engineering*, vol. 2024, no. 1, 3909173, 2024.

[18] M. Wang *et al.*, "Learn-IDS: Bridging gaps between datasets and learning-based network intrusion detection," *Electronics*, vol. 13, no. 6, 1072, 2024.

[19] A. Khacha *et al.*, "Robust intrusion detection for IoT networks: An integrated CNN-LSTM-GRU approach," in *Proc. 2023 International Conf. on Networking and Advanced Systems (ICNAS)*, 2023, pp. 1–6.

[20] D. Kilichev, D. Turimov, and W. Kim, "Next–generation intrusion detection for IoT EVCS: Integrating CNN, LSTM, and GRU models," *Mathematics*, vol. 12, no. 4, 571, 2024.

[21] C. Hazman *et al.*, "A smart model integrating LSTM and XGBoost for improving IoT-enabled smart cities security," *Cluster Comput.*, vol. 28, 70, 2024.

[22] N. Kumar and S. Sharma, "A hybrid modified deep learning architecture for intrusion detection system with optimal feature selection," *Electronics*, vol. 12, no. 19, 4050, 2023.

[23] A. F. Otoom, W. Eleisah, and E. E. Abdallah, "Deep learning for accurate detection of brute force attacks on IoT networks," *Procedia Computer Science*, vol. 220, pp. 291–298, 2023.

[24] R. Alghamdi and M. Bellaiche, "An ensemble deep learning based IDS for IoT using lambda architecture," *Cybersecurity*, vol. 6, 5, 2023.

[25] E. H. Salman *et al.*, "An anomaly intrusion detection for high-density internet of things wireless communication network based deep learning algorithms," *Sensors*, vol. 23, no. 1, 206, 2023.

[26] Z. Wang *et al.*, "A lightweight intrusion detection method for IoT based on deep learning and dynamic quantization," *PeerJ Comput. Sci*., vol. 9, e1569, 2023.

[27] X. Yang *et al.*, "An enhanced intrusion detection system for IoT networks based on deep learning and knowledge graph," *Security and Communication Networks*, vol. 2022, no. 1, 4748528, 2022.

[28] K. Bella *et al.*, "An efficient intrusion detection system for IoT security using CNN decision forest," *PeerJ Comput. Sci*., vol. 10, e2290, 2024.

[29] A. Henry *et al.*, "Composition of hybrid deep learning model and feature optimization for intrusion detection system," *Sensors*, vol. 23, no. 2, 890, 2023.

[30] P. Sanju, "Enhancing intrusion detection in IoT systems: A hybrid metaheuristics-deep learning approach with ensemble of recurrent neural networks," *Journal of Engineering Research*, vol. 11, no. 4, pp. 356–361, 2023.

[31] R. Anushiya and V. S. Lavanya, "A new deep-learning with swarm based feature selection for intelligent intrusion detection for the Internet of things," *Measurement: Sensors*, vol. 26, 100700, 2023.

[32] R. Chaganti *et al.*, "Deep learning approach for SDN-enabled intrusion detection system in IoT networks," *Information*, vol. 14, no. 1, 41, 2023.

[33] D. Z. Rodríguez *et al.*, "Attentive transformer deep learning algorithm for intrusion detection on IoT systems using automatic Xplainable feature selection," *PLOS One*, vol. 18, no. 10, e0286652, 2023.

[34] H. Kim *et al.*, "A transferable deep learning framework for improving the accuracy of internet of things intrusion detection," *Future Internet*, vol. 16, no. 3, 80, 2024.

[35] C. Mahjoub *et al.*, "An adversarial environment reinforcement learning-driven intrusion detection algorithm for internet of things," *EURASIP Journal on Wireless Communications and Networking*, vol. 2024, 21, 2024.

[36] C. Hazman *et al.*, "Enhanced IDS with deep learning for IoT-based smart cities security," *Tsinghua Science and Technology*, vol. 29, no. 4, pp. 929–947, 2024.

[37] H. Shah *et al.*, "Deep learning-based malicious smart contract and intrusion detection system for IoT environment," *Mathematics*, vol. 11, no. 2, 418, 2023.

[38] A. Awajan, "A novel deep learning-based intrusion detection system for IoT networks," *Computers*, vol. 12, no. 2, 34, 2023.

[39] M. G and P. Maheswaravenkatesh. (January 2024). A lightweight convolutional neural network based network intrusion detection and classification method for social internet of things. *Research Square.* [Online]. Available: https://www.researchsquare.com/article/rs-3795283/v1

[40] J. Jose and D. V. Jose, "AS-CL IDS: Anomaly and signature-based CNN-LSTM intrusion detection system for internet of things," *Int. J. Adv. Technol. Eng. Explor*., vol. 10, no. 109, pp. 1–18, 2024.

[41] R. Devendiran and A. V. Turukmane, "Dugat-LSTM: Deep learning based network intrusion detection system using chaotic optimization strategy," *Expert Syst. Appl*., vol. 245, 123027, 2024.

[42] C. Hazman *et al.*, "Intrusion detection framework for IoT-based smart environments security," in *Proc. International Conf. on Artificial Intelligence and Smart Environment*, 2023, pp. 546–552.

[43] A. A. Wardana *et al.*, "Ensemble averaging deep neural network for botnet detection in heterogeneous internet of things devices," *Sci. Rep*., vol. 14, 3878, 2024.

[44] S. Abbas *et al.*, "Evaluating deep learning variants for cyber-attacks detection and multi-class classification in IoT networks," *PeerJ Comput. Sci*., vol. 10, e1793, 2024.

[45] B. Sharma *et al.*, "Explainable artificial intelligence for intrusion detection in IoT networks: A deep learning based approach," *Expert Syst. Appl*., vol. 238, 121751, 2024.

[46] G. Parimala and R. Kayalvizhi, "Improved elman deep learning model for intrusion detection system in internet of things," *J. Internet Serv. Inf. Secur*., vol. 14, no 1, pp. 121–137, 2024.

[47] A. Nazir *et al.*, "A deep learning-based novel hybrid CNN-LSTM architecture for efficient detection of threats in the IoT ecosystem," *Ain Shams Eng. J.,* vol. 15, no. 7, 102777, 2024.

[48] F. H. Zawaideh *et al.*, "Intrusion detection system for (IoT) networks using convolutional neural network (CNN) and XGBOOST algorithm," *Journal of Theoretical and Applied Information Technolog,* vol. 102, no. 4, pp. 1750–1759, 2024.

[49] A. S. Dina, A. B. Siddique, and D. Manivannan, "A deep learning approach for intrusion detection in internet of things using focal loss function," *Internet Things*, vol. 22, 100699, 2023.

[50] M. Catillo, A. Pecchia, and U. Villano, "A deep learning method for lightweight and cross-device IoT botnet detection," *Appl. Sci.,* vol. 13, no 2, 837, 2023.

[51] N. W. Khan *et al.*, "A hybrid deep learning-based intrusion detection system for IoT networks," *Math. Biosci. Eng.,* vol. 20, no. 8, pp. 13491–13520, 2023.

[52] R. Lazzarini, H. Tianfield, and V. Charissis, "A stacking ensemble of deep learning models for IoT intrusion detection," *Knowl.-Based Syst.,* vol. 279, 110941, 2023.

[53] M. Keshk *et al.*, "An explainable deep learning-enabled intrusion detection framework in IoT networks," *Inf. Sci.,* vol. 639, 119000, 2023.

[54] S. A. Bakhsh *et al.*, "Enhancing IoT network security through deep learning-powered intrusion detection system," *Internet Things,* vol. 24, 100936, 2023.

[55] S. Soliman, W. Oudah, and A. Aljuhani, "Deep learning-based intrusion detection approach for securing industrial internet of things," *Alex. Eng. J.,* vol. 81, pp. 371–383, 2023.

[56] A. Odeh and A. A. Taleb, "Ensemble-based deep learning models for enhancing IoT intrusion detection," *Appl. Sci.,* vol. 13, no. 21, 11985, 2023.

[57] O. Elnakib *et al.*, "EIDM: Deep learning model for IoT intrusion detection systems," *J. Supercomput.,* vol. 79, pp. 13241–13261, 2023.

[58] S. Baniasadi *et al.*, "A novel deep supervised learning-based approach for intrusion detection in IoT systems," *Sensors,* vol. 22, no. 12, 4459, 2022.

[59] E. H. Qazi *et al.*, "An intelligent and efficient network intrusion detection system using deep learning," *Comput. Electr. Eng.,* vol. 99, 107764, 2022.

[60] I. Tareq *et al.*, "Analysis of ToN-IoT, UNW-NB15, and edge-IIoT datasets using DL in cybersecurity for IoT," *Appl. Sci.,* vol. 12, no. 19, 9572, 2022.

[61] B. Sharma *et al.*, "Anomaly based network intrusion detection for IoT attacks using deep learning technique," *Comput. Electr. Eng.,* vol. 107, 108626, 2023.

[62] T. Saba *et al.*, "Anomaly-based intrusion detection system for IoT networks through deep learning mode," *Comput. Electr. Eng.,* vol. 99, 107810, 2022.

[63] A. Halbouni *et al.*, "CNN-LSTM: Hybrid deep neural network for network intrusion detection system," *IEEE Access,* vol. 10, pp. 99837–99849, 2022.

[64] A. M. Banaamah and I. Ahmad, "Intrusion detection in IoT using deep learning," *Sensors,* vol. 22, no. 21, 8417, 2022.

[65] J. Jose and D. V. Jose, "Deep learning algorithms for intrusion detection systems in internet of things using CIC-IDS 2017 dataset," *Int. J. Electr. Comput. Eng.,* vol. 13, no 1, pp. 1134–1141, 2023.

[66] K. O. A. Alimi *et al.*, "Refined LSTM based intrusion detection for denial-of-service attack in internet of things," *J. Sens. Actuator Netw.,* vol. 11, no. 3, 32, 2022.

[67] M. Vishwakarma and N. Kesswani, "DIDS: A deep neural network based real-time intrusion detection system for IoT," *Decis. Anal. J.,* vol. 5, 100142, 2022.

[68] R. Ahmad *et al.*, "A comprehensive deep learning benchmark for IoT IDS," *Computers & Security*, vol. 114, 102588, 2022.

[69] S. Balaji *et al.*, "A GAN-based hybrid deep learning approach for enhancing intrusion detection in IoT networks," *IJACSA,* vol. 15, no. 6, pp. 348–354, 2024.

[70] R. Elsayed *et al.*, "A hierarchical deep learning-based intrusion detection architecture for clustered internet of things," *Journal of Sensor and Actuator Networks,* vol. 12, no. 1, 3, 2023.

[71] R. Priyanka *et al.*, "A hybrid cluster based intelligent IDS with deep belief network to improve the security over wireless sensor network," *International Journal of Intelligent Systems and Applications in Engineering,* vol. 12, no. 17S, pp. 225–238, 2024.

[72] G. Kocher and G. Kumar, "A hybrid deep learning approach for effective intrusion detection systems using spatial-temporal features," *Adv. Eng. Sci.,* vol. 54, no. 02, pp. 1503–1519, 2022.

[73] Y. A. Sawafi, A. Touzene, and R. Hedjam, "Hybrid deep learning-based intrusion detection system for RPL IoT networks," *Journal of Sensor and Actuator Networks,* vol. 12, no. 2, 21, 2023.

[74] T. Yang *et al.*, "A lightweight intrusion detection algorithm for IoT based on data purification and a separable convolution improved CNN," *Knowledge-Based Systems,* vol. 304, 112473, 2024.
[75] I. Idrissi, M. Azizi, and O. Moussaoui, "A lightweight optimized deep learning-based host-intrusion detection system deployed on the edge for IoT," *International Journal of Computing and Digital Systems,* vol. 11, no. 1, pp. 209–216, 2022.
[76] S. Hizal, U. Cavusoglu, and D. Akgun, "A novel deep learning-based intrusion detection system for IoT DDoS security," *Internet of Things,* vol. 28, 101336, 2024.
[77] M. Maheswari and R. A. Karthika, "A novel hybrid deep learning framework for intrusion detection systems in WSN-IoT networks," *Intelligent Automation & Soft Computing,* vol. 33, no. 1, pp. 365–382, 2022.
[78] B. Raviprasad *et al.*, "Accuracy determination using deep learning technique in cloud-based IoT sensor environment," *Measurement: Sensors,* vol. 24, 100459, 2022.
[79] M. M. Rashid *et al.*, "Adversarial training for deep learning-based cyberattack detection in IoT-based smart city applications," *Computers & Security,* vol. 120, 102783, 2022.
[80] J. Cao *et al.*, "An efficient deep learning approach to IoT intrusion detection," *The Computer Journal,* vol. 65, no. 11, pp. 2870–2879, 2022.
[81] H. Mohamed, A. Hamza, and H. Hefny, "An efficient intrusion detection approach using ensemble deep learning models for IoT," *International Journal of Intelligent Engineering & Systems,* vol. 16, no. 1, pp. 350–363, 2022.
[82] M. A. Alsoufi *et al.*, "Anomaly-based intrusion detection model using deep learning for IoT networks," *Computer Modeling in Engineering & Sciences,* vol. 141, no. 1, pp. 823–845, 2024.
[83] M. Nobakht, R. Javidan, and A. Pourebrahimi, "DEMD-IoT: A deep ensemble model for IoT malware detection using CNNs and network traffic," *Evolving Systems,* vol. 14, pp. 461–477, 2023.