Shahid Bahonar University of Kerman
Journal of Mahani Mathematical Research
Print ISSN: 2251-7952 Online ISSN: 2645-4505
Journal of Mahani Mathematical Research JMMR 2012

DEEP LEARNING-BASED INTRUSION DETECTION SYSTEMS: A COMPREHENSIVE SURVEY OF FOUR MAIN FIELDS OF CYBER SECURITY

R. Jafari Gohari, L. Aliahmadipour, and M. Kuchaki Rafsanjani ✉

Article type: Research Article

(Received: 25 July 2022, Received in revised form 18 September 2022)

(Accepted: 15 November 2022, Published Online: 16 November 2022)

Abstract. The security flaws in cyber security have always put the users and organizations at risk, which as a result created catastrophic conditions in the network that could be either irreversible or sometimes too costly to recover. In order to detect these attacks, Intrusion Detection Systems (IDSs) were born to alert the network in case of any intrusions. Machine Learning (ML) and more prominently deep learning methods can be able to improve the performance of IDSs. This article focuses on IDS approaches whose functionalities rely on deep learning models to deal with the security issue in Internet of Things (IoT), wireless networks, Software Defined Networks (SDNs), and Industrial Control Systems (ICSs). To this, we examine each approach and provide a comprehensive comparison and discuss the main features and evaluation methods as well as IDS techniques that are applied along with deep learning models. Finally, we will provide a conclusion of what future studies are possibly going to focus on in regards to IDS, particularly when using deep learning models.

Keywords: Intrusion Detection System (IDS), Cyber Security, Deep Learning, Internet of Things (IoT), Wireless Network, Software Defined Network (SDN), Industrial Control System (ICS).
2020 MSC: Primary 68T07, 68M25, 68M10.

1. Introduction

Cyber security threats and the risk of either losing confidential data or irreversible disruption in the network have always been taken seriously by security researchers in order to detect the attacks in the network if not first prevented. This could simply mean privileged access from an authorized user [44], disturbances in the network, exploiting the vulnerability of users or devices, or even bringing down an infrastructure by implementing Distributed Denial of Service (DDoS) attacks [87]. That is why security research for such a crucial phenomenon has been going on for years in the ever-changing landscape of cyber security, and as a result of that, the growth for diverse solutions has always

✉ kuchaki@uk.ac.ir, ORCID: 0000-0002-3220-4839
DOI: 10.22103/jmmr.2022.19961.1305
Publisher: Shahid Bahonar University of Kerman
How to cite: R. Jafari Gohari, L. Aliahmadipour, M. Kuchaki Rafsanjani, *Deep learning-based intrusion detection systems: A comprehensive survey of four main fields of cyber security*, J. Mahani Math. Res. 2023; 12(2): 289-324.

been increasing in all sorts of networks. Therefore, gathering all the information and research that is in our disposal can lead to a better understanding of what path is this landscape pursuing.

Intrusion Detection Systems (IDSs) are regarded as one of the most ideal solutions in cyber security research due to being an all-inclusive method against almost all sorts of attacks in the network and as a result, we can witness IDS approaches in mainly different fields of cyber security including industrial systems or even wireless networks. That is why IDSs are believed to be an indispensable part of the network when it comes to security and reliability.

Looking at the bigger picture can help us better comprehend a solution like an IDS if all the solutions are put together in order for a comparison to be made so that this huge landscape and its future trends can be more easily understood. Different IDS-based solutions have been provided, which on one hand can be categorized as Host Intrusion Detection Systems (HIDS) and Network Intrusion Detection Systems (NIDS) [24]. As the name suggests, HIDS is more capable to enhance the security of a single system while on the other hand NIDS is meant to secure an entire network.

On the other hand, there are signature-based IDS approaches [46] that have been proposed in order to detect attacks based on earlier events, hence, the greatest disadvantage of these types of IDSs is that they are prone to inefficiency and lack of awareness when it comes to zero-day attacks. Due to this simple fact, anomaly-based approaches [86] have been put forward to tackle this challenge, which when coupled with machine learning techniques, the IDS will be capable of learning the normal behavior and as a result will be able to detect abnormal behavior anytime there is an anomaly or unrecognized pattern throughout the network.

There are also other works that took one step further to provide a more secure IDS. Hoque et al. [29] were among those who utilized evolutionary algorithms such as Genetics algorithm to innovate a new way for detecting anomalies in the network. Moreover, Mohammadi et al. [67] implemented a new IDS that utilizes feature selection mechanisms to deal with massive data in the network.

As it can be seen, the vastness of machine learning techniques and choosing the right method can be a bit of a hassle considering the security criteria in each approach. Machine learning models can be categorized into supervised, semi-supervised, and unsupervised learning. While all the data in supervised machine learning models are labeled, the training data in semi-supervised models contain very few labeled examples and a large number of unlabeled examples. For unsupervised learning, there is no labeled data whatsoever and the model has the responsibility of finding the relationship between the data [85].

In spite of all these machine learning models, there is still an urging need for more accurate models that can outperform other traditional approaches. That is why deep learning techniques have gained popularity among researchers in order to acquire higher accuracy as well as higher detection rates. Some of the most popular deep learning approaches can be Convolutional Neural Network (CNN), Auto-Encoder (AE) models , Deep Belief Network (DBN) and Recurrent Neural Network (RNN). Deep learning methods as a substantial feature in an IDS can be used not only for the purpose of classification but also it can be utilized for feature extraction as well to determine what features stand out in the dataset.

Choosing the appropriate method is one thing and understanding the security issues in the network is another. In other words, a network can range from a simple wireless network with a couple of devices to ultimately a large-scale industrial network where hundreds or even thousands of devices are interconnected, which as a consequence makes it potentially a target for all sorts of attacks. Therefore, many IDS approaches have been presented to be efficient in such networks using deep learning approaches. That is why we believe taking a deep dive into all the deep learning approaches in different fields of cyber security can help us determine the future trends of IDSs.

In this work, we are going to provide a comparative survey on IDS approaches in four main fields of cyber security, namely Internet of Things (IoT), wireless networks, Software Defined Networks (SDNs) and also Industrial Control Systems (ICSs). We will provide details about approaches in the aforementioned fields that utilize deep learning algorithms either for classification purposes or for feature encoding or feature selection procedure. Moreover, we will discuss all the approaches in detail and talk about the simulations they have used as well as their evaluation metrics and the datasets that they have trained the model with. In addition, we will compare the accuracy of each approach in its own field and discuss the advantages and disadvantages of each approach as well.

The main contributions of our work are as follows:

- Providing a comprehensive overview of proposed deep learning based IDS approaches in the four main fields of cyber security, namely IoT, wireless networks, SDN environment as well as ICSs.
- Numerous works investigated IDSs that were signature-based or anomaly-based as well as stateful IDS [61]. Khraisat et al. investigated different

types of attacks based on intrusion methods [52]. Ferrag et al. provided their taxonomy to investigate different approaches in comparison to similar works [20]. Aldweesh et al. [4] and Aleesa et al. [5] investigated new approaches of IDS when it comes to detection mechanisms as well as input data and innovative ways of detection intrusions in the system. The authors' perspective is not concentrated on one specific field, although a great set of attributes are provided in their investigations. That is why one of the most important contributions in this work is concentration on a more comprehensive perspective in regards to the four specific fields for investigating approaches with more maneuverability and more focused vision.

- Describing what each approach has accomplished in addition to the architecture or system model that they propose.
- Providing details of each field separately in a table that includes important attributes of the proposed approach. These attributes are simulation environment ,being a real time or not, the approach classifier, dataset, feature engineering/selection method, evaluation metrics and accuracy.
- Providing a taxonomy of our work as well as a taxonomy of overall machine learning approaches, including deep-learning and non-deep learning based algorithms.
- Giving a final conclusion of our research and the recommendations that can help the future researches that are related to deep learning based IDSs.

The following sections of our work are as follows: In Section 2, we will discuss our research methodology where we talk about the steps that we took to complete our review work. In Section 3 we discuss the preliminaries and the most prominent deep learning approaches as well as the most important datasets that were used in the investigated works. In Section 4, we present our review where we will examine the approaches and provide details of each approach in its own specified field, which are discussed in detail, and finally, in Section 5, we provide our final conclusion in regards to each field.

2. Research Methodology

For conducting this review, we have taken certain steps to make sure that the purpose of this literature is fulfilled. These steps are shown in Figure 1 and as it can be seen, the first step is to understand what other perspectives regarding security IDSs have been proposed. In this way, we could understand what other researchers have been able to achieve as well as be able to understand what gaps exist in terms of deep learning-based IDSs and machine learning.

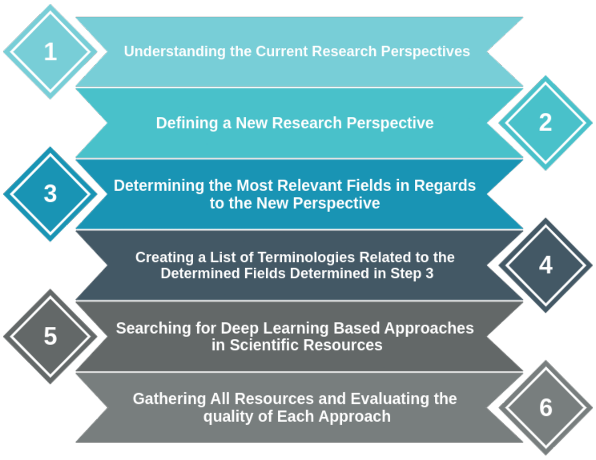
1
Understanding the Current Research Perspectives
2
Defining a New Research Perspective
3
Determining the Most Relevant Fields in Regards to the New Perspective
4
Creating a List of Terminologies Related to the Determined Fields Determined in Step 3
5
Searching for Deep Learning Based Approaches in Scientific Resources
6
Gathering All Resources and Evaluating the quality of Each Approach

Figure 1. The research steps taken to gather the necessary approaches.

The second step that we took was filling the gap within the current IDS research perspectives. The new view that we defined simply did not exist. Finding a new perspective meant understanding other previous surveys. As described in the introduction section, the new perspective was defined based on a long but completely thought-out process in order to be thorough in the investigation of new approaches.

Now that we had a new perspective, the third and fourth step involved examining the most relevant works that proposed new approaches to IDS using machine learning. The most important fields that were chosen were IoT, wireless networks, SDNs and finally ICSs. In addition, we also used a general set of phrases to to get our hands on the most relevant approaches regardless of its algorithm. Those phrases are as follows:

- Intrusion detection internet of things.
- Intrusion detection IoT.
- Intrusion detection wireless Networks.
- Intrusion detection MANET.
- Intrusion detection mobile ad-hoc network.
- Intrusion detection VANET.
- Intrusion detection vehicular ad-hoc network.
- Intrusion detection SDN
- Intrusion detection software defined network.
- Intrusion detection ICS
- Intrusion detection industrial control system.

After finishing the third and fourth steps, we moved to the fifth step and found all the articles we needed in order to conduct the survey. Countless articles were found that seemed irrelevant during this process and we needed to make sure the most qualified articles were used for this survey. Therefore, the sixth step helped us qualify all the relevant approaches and conduct the survey in the most fruitful way possible.

To see which work was relevant and more important than others, with the help of following fundamental questions, we have been able to filter the qualified works:

- Is the approach solving the IDS-related problem through either a novel or an innovative algorithm?
- Is the approach trying to solve the IDS-related security flaws through deep learning algorithms?
- Has the approach been able to achieve sufficient accuracy using deep learning models?
- Has the approach involved data generation so they can create a new dataset in the four aforementioned fields of cybersecurity?
- Intrusion detection mobile ad-hoc network.
- Has the approach been able to use the latest tools and simulations to make sure the final result could be potentially used in the real world scenarios?

We have conducted this survey on the approaches that have been carried out in the last five years in either of four fields of our interest. Figure 2 illustrates the year and number of works published in each of the last five years.

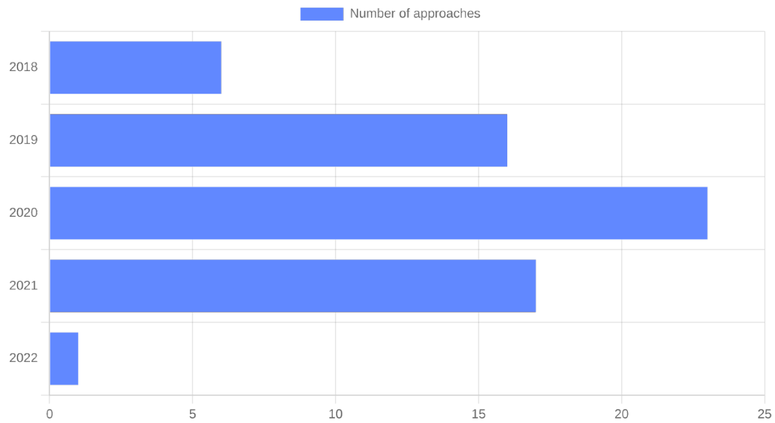

Figure 2. The number of IDS approaches in each year.

3. Preliminaries

This part of our work will take a better look at types of IDS as well as deep learning approaches and also the most important datasets that have been used by the investigated IDS approaches.

3.1. **Deep Learning Algorithms.** Many algorithms have been around for a long time to perform simple tasks of Artificial Intelligence (AI), algorithms such as Multilayer Perceptron (MLP) and Artificial Neural Networks (ANNs). But more advanced algorithms emerged with the emergence of data. Deep learning is a derivative of machine learning algorithms that can be regarded as one of the most performant and reliable learning algorithms that can perform classification tasks more efficiently. Deep Neural Networks (DNN) usually contain more than one layer, and each layer of course has neurons. Each neuron is affected by factors such as weight and bias as well as the activation function. This is the basic structure of all deep learning algorithms. Our taxonomy in Figure 3 illustrates deep and non deep-learning algorithms that driven by machine learning. As it can be seen, different algorithms are divided into different categories. The outcome of all these algorithms may be dependent on some factors such as the quality of the data, data preparation and data normalization as well as using the correct algorithm in its own domain since all of them are not proposed to be efficient in every situation. Figure 4 demonstrates the taxonomy of deep learning algorithms that are used among the investigated approaches. The IDS approaches are categorized into four fields of IoT, SDN, wireless and ICS networks. In addition, hybrid approaches that propose a deep learning-based IDS in a hybrid environment are also included in the taxonomy.

As it can be seen, different algorithms are divided into different categories. The outcome of all these algorithms may be dependent on some factors such as the quality of the data, data preparation and data normalization as well as using the correct algorithm in its own domain since all of them are not proposed to be efficient in every situation. Figure 4 demonstrates the taxonomy of deep learning algorithms that are used among the investigated approaches. The IDS approaches are categorized into four fields of IoT, SDN, wireless and ICS networks. In addition, hybrid approaches that propose a deep learning-based IDS in a hybrid environment are also included in the taxonomy.

Each auto-encoder has two steps: an encoder for mapping the input data into the code, and a decoder for constructing input data from the code. More specifically, auto-encoders support unsupervised learning of dataset encoding for dimensionality reduction, by training the network to ignore the signal noise.

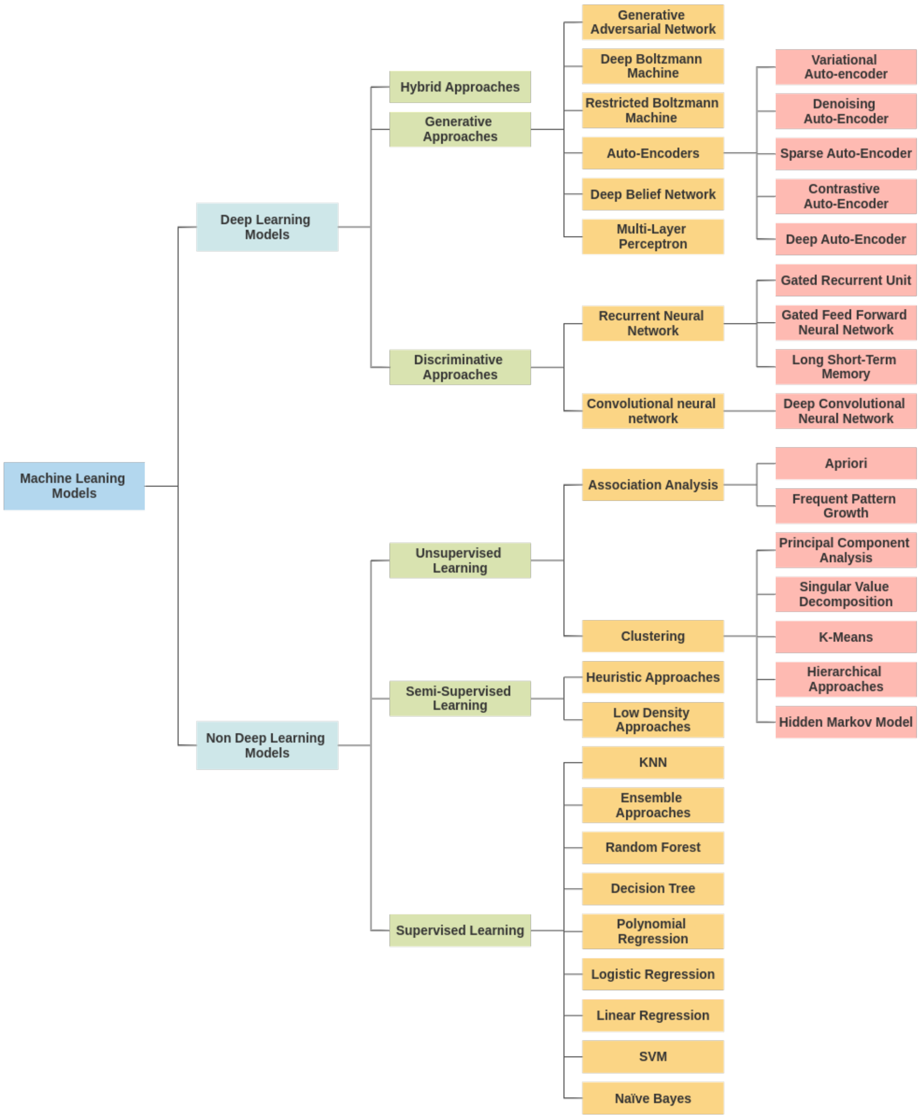

Figure 3. Complete taxonomy of deep and machine learning algorithms.

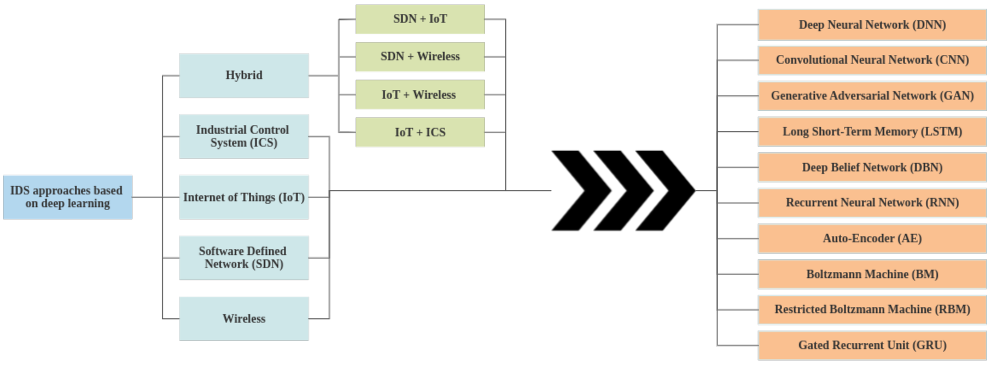
IDS approaches based on deep learning
Hybrid
Industrial Control System (ICS)
Internet of Things (IoT)
Software Defined Network (SDN)
Wireless
SDN + IoT
SDN + Wireless
IoT + Wireless
IoT + ICS
Deep Neural Network (DNN)
Convolutional Neural Network (CNN)
Generative Adversarial Network (GAN)
Long Short-Term Memory (LSTM)
Deep Belief Network (DBN)
Recurrent Neural Network (RNN)
Auto-Encoder (AE)
Boltzmann Machine (BM)
Restricted Boltzmann Machine (RBM)
Gated Recurrent Unit (GRU)

Figure 4. Taxonomy of deep learning algorithms in the investigated approaches.

DBNs: In deep learning, a consecutive stacked Restricted Boltzmann machine (RBM) creates DBN [27] whose characteristics create a generative model that are probabilistic as well. The result of a DBN network addresses the ANN problems with training data. Moreover, the DBN network deals with local minimum problems as well as the problem of enormous datasets and slow-going training. The utilization of such a network can be in image recognition, Natural Language Processing (NLP) as well as intrusion detection.

RNNs: ANNs coupled with feed-forwarding capability is considered to be RNN [74]. The architecture of an RNN allows the outcome of the input layer to be unidirectionally connected to the hidden layers. On the other hand, hidden layers are connected to the next layer and to the hidden layers themselves. This helps RNN to be efficiently used in areas such as security and malware detection since the RNNs are capable of maintaining the current and previous input. This increases the detected rate since the probability of the detection is based on the current and previous input.

GAN: ANNs are used in the architecture of numerous algorithms. Generative Adversarial Network (GAN) [23] is also one of them which utilizes dual ANN alongside each other. Most use cases for GAN are image classification, NLP, translation including text-to-image, and image-to-image translation.

CNNs: Another algorithm that utilizes ANN with multiple hidden layers is the CNN algorithm. The most prominent use case for CNN algorithm is image classification [88]. However, this algorithm can be used within the security domain as well, one of the most prevalent uses of which is intrusion detection. In addition, this algorithm can be used for feature-engineering purposes. CNN works with 2D-matrix and provides the final result based on assigning importance to various parts of the data.

LSTM: Long Short-Term Memory (LSTM) [28] is part of the RNN algorithm that is capable of learning order dependence. Another advantage is the fact that LSTM can overcome phenomena like vanishing gradient and exploding gradients during the training phase. Most important fields that LSTM has a lot of use cases in are grammar learning, speech recognition and time-series prediction.

Boltzmann Machine: Boltzmann Machines (BM) are comprised of RNNs whose final decisions are in binary [63]. When different Boltzmann machines are put together, DBNs are created. An interesting usage of BM is discovering complex patterns and extracting interesting features. This system is given a binary set of vector data as input. The system continuously updates its weights as each feature is processed. Other use cases of Boltzmann machines are in search problems and optimizations.

Auto-Encoders: The way AEs work [51] is basically by implementing dimensionality reduction, which allows the algorithm to learn by ignoring noise in the data. AEs have an input, hidden and output layer. This algorithm uses backpropagation to continuously train itself. Moreover, the algorithm sets the target output values to equal the inputs. In this way, the encoding layers are obligated to utilize dimensionality reduction and hence remove noise.

3.2. **IDS Datasets.** Several important IDS datasets are employed by the investigated deep learning-based IDS approaches In this section, we will discuss the major IDS datasets to get a better understanding of their details as well as the reason why so many IDS-based approaches lean towards these datasets.

NSL-KDD: NSL-KDD was created to address the inefficiency of KDD'99 dataset [40]. The dataset is considered more comprehensive compared to other available datasets when it comes to network-based IDSs. The number of records is sufficiently high in the dataset, which makes it an advantage when it comes to a reliable source in experimental environments. Redundant records were added in this dataset, and as a result, we can see in the investigated methods that different machine learning results vary using the same dataset.

KDD-CUP-99: This dataset was introduced by DARPA in 1998, whose purpose was to survey and assess research within the field of IDS [32]. This dataset contains a variety of network intrusions such as DOS attacks, R2L attack, U2R attack and Probing attacks. In addition, the dataset provides records of individual TCP connections as well as content features within a connection inside a domain and also traffic features using a two-second time window.

CICDDoS2019: This dataset has the analysis of the final result of network traffic, all of which are labeled according to the timestamps, source IP, destination IPs, source port, destination port, protocols and finally the attack type [37]. The dataset file is presented in a Comma Separated File (CSV). The characteristics of an attack of 25 users were built in this dataset according to HTTP, HTTPS, FTP, SSH, and Email protocols. In terms of DDoS.

UNSW-NB15: UNSW-NB15 dataset, which sometimes is referred to as UNSQ-NB15 dataset, has nine various attacks [68], which are accordingly; Fuzzers, Analysis, Backdoors, DoS, Exploits, Generic, Reconnaissance, Shellcode and Worms. One interesting fact about this dataset is the fact that it contains raw network packets. The number of records in the training set is 175,341 records and the testing set is 82,332 records, all of which have two labels; attack and normal.

CICIDS2017: This dataset also has benign and up-to-date attacks, records of the dataset are timestamp, source, and destination IPs, source and destination ports, protocols, and attack [38], which as a result is presented in a CSV file for. The capturing of the data took 5 days and the attacks that are included in this dataset are Brute Force FTP, Brute Force SSH, DoS, Heartbleed, Web Attack, Infiltration, Botnet, and DDoS.

IoT-23: IoT-23 is a new dataset which was collected from various IoT devices [36]. It contains 20 malware captures executed in IoT devices, and 3 captures for benign IoT devices traffic. The first appearance of the dataset was in January 2020, which showed the captures that were collected between 2018 and 2019. This IoT network traffic was captured in the Stratosphere Laboratory, AIC group, FEL, CTU University, Czech Republic. The purpose of this big dataset is to present a real-world like IoT malware dataset with label data in order for researchers to develop machine learning models and new algorithms.

N-BaIoT: This dataset [31] was originally created to differentiate between benign and malicious traffic data when it comes to intrusion and anomaly detection. Interestingly, according to the authors, one view can be regarded in such a way that the dataset can be divided into 10 attacks which takes place by 2 botnets, and yet in another view, the dataset can be used for multi classification; 10 classes of attacks in addition to a single benign class.

Bot-IoT: The creation of the Bot-Iot dataset was done by the Cyber Range Lab of UNSW Canberra [34] in order to design a realistic network environment. Various formats of the datasets are provided including the pcaps file and CSV file. It contains 72.000.000 records, which include DDoS, DoS, OS and Service Scan, Keylogging and Data exfiltration attacks, with the DDoS and DoS attacks further organized, based on the protocol used.

InSDN Dataset: When it comes to a comprehensive dataset that contains records of SDN-based environments, InSDN becomes a good choice. This dataset, which was proposed by Elsayed et al. [33] can offer SDN-based attack specific records that are publicly available to simulate an IDS for SDN networks. It contains the benign and different attack categories that can carry out under different circumstances of the SDN environment.

CSE-CIC-IDS2018: This dataset uses [39] the idea of profiles in order to create a dataset that can have descriptions of intrusions as well as abstract distribution models for applications, protocols, or lower level network entities. The profiles can be utilized by agents to create events on the network. Protocols that were taken into consideration in the dataset are HTTPS, HTTP, SMTP, POP3, IMAP, SSH, and FTP.

Gas Pipeline Dataset: The dataset was created due to the lack of appropriate datasets in networks based on ICS. The dataset was generated at the time of the logging of the network [35] traffic of the laboratory Supervisory Control and Data Acquisition (SCADA) systems. This dataset contains the records of various attacks, including Naive Response Injection, Complex Response Injection, State Command Injection, Parameter Command Injection, Function Code Injection, Denial of Service, and Reconnaissance.

4. Investigated deep learning-based IDS Approaches

4.1. **Deep learning-based IDS for wireless Environment.** This section covers all the approaches that have proposed a deep learning-based IDS in wireless networks, whether as a classifier or as a feature extraction algorithm. The information regarding these approaches are in Table 1. As it can be seen, the first three columns respectively show the name of the authors, the tools they utilized and whether their approach is a real-time implementation of IDS. The next three columns illustrate the classification algorithm, the dataset and the feature engineering approach they used. And the final two columns demonstrate evaluation metrics and the accuracy of the approach.

Kasongo and Sun [48] proposed a method that utilizes a Deep Gated Recurrent Unit (DGRU) that is used as the classifier of the IDS. They compare their framework with LSTM and Random Forest and Feed Forward Deep Neural Networks as well as Naive Bayes (NB). They used Python and one of its main libraries, Keras to implement such a framework. Accuracy, recall, F1-measure and precision are used as evaluation metrics of the framework and the final result demonstrated an 88.42% accuracy.

Riyaz and Ganapathy [75] were concerned with the feature extraction methods since they believed this step improves the model's efficiency and ultimately increases the detection accuracy of IDS. Their model is denoted as Conditional Random Field-Linear Correlation Coefficient-Based (CRF-LCFS), which is fundamentally based upon CNN algorithm and interestingly for the classification of the model they have also utilized CNN algorithm. The combination of the two demonstrated the lowest False Alarm Rate (FAR). Their approach was implemented using Python and Tensorflow libraries which yielded a 98.88% accuracy by extracting features from the KDD-CUP-99 dataset.

Kasongo and Sun [49] propose a deep LSTM model which uses the NSL-KDD dataset. The evaluation metrics in the model are accuracy, precision, recall, and F1 score. The model is compared with traditional machine learning models such as Support Vector Machine (SVM), Random Forest, and Naive

Bayes. The model accuracy was 99.51% using only manual encoding techniques, which means that compared to other IDS approaches, this approach sufficed to determine the most important features via human interpretation.

Gowdhaman and Dhanapal [25] compared their model with SVM algorithm, Decision Tree, and Random Forest algorithms in different scenarios under various attack simulations, which as a result illustrated a significant increase in the accuracy of the model. They used MATLAB for their experiments and their approach yielded 95.53% accuracy. Their chosen evaluation methods were accuracy, recall, precision, F1-score as well as False Positive Rate (FPR) and False Negative Rate (FNR).

The work of Duan et al. [15] compared to other investigated IDS approaches within the field of wireless shows that they have utilized a rather different dataset, AWID, which is mainly used for Wi-Fi networks. They compared their model with Random Forest, Naive Bayes, Random Tree, and J48 algorithms and as it turned out their model yields the best result when it comes to injection attacks. Their work demonstrated a 99% accuracy on average in four different attack scenarios.

Sbai and Elboukhari's approach [77] is implemented inside a MANET where CICDDoS2019 is used as the main dataset. Their approach tests two DNNs, and they plan to implement the same model for different attacks in the future. They also used NS-3 as the main simulation environment, in which they yield a 99.99% accuracy, which is extremely efficient when manual feature encoding technique is used to select the desired features from the dataset.

Dilipkumar and Durairaj [14] implemented their approach for creating a robust IDS is denoted as Centrality Epilson Greedy Swarm and Gradient Deep Belief Classifier (CEGS-GDBC) since DBN is coupled with a clustering method called Epilson Greedy Swarm Optimization to be used against network intrusions such as DDoS attacks. Their work was implemented inside a MANET area, in which they used MATLAB and NS-2 as their main simulation tools. Their evaluation metrics were attack detection rate, memory consumption as well as the time of computation for identifying and isolating the attacker. Their main classifier was gradient DBN but that is not the only part of their algorithm. The performance shows their approach has the highest detection rate as well as the lowest computational time and memory consumption.

Huang and Lei [41] proposed an IDS approach for Ad-hoc networks that utilize GAN that is capable of dealing with imbalanced datasets. The main feature extraction algorithm in this IDS is Feed Forward Neural Network (FNN). In other words, transforming raw network features into feature vectors. Some of the algorithms that have been used as classifiers to compare the proposed model are MLP, Random Forest, CNN, and SVM. Their method uses three datasets, namely NSL-KDD, UNSW-NB15 and CICIDS2017, which as a result the approaches reaches 84.45%, 82.53, and 99.79% accuracy for each dataset respectively.

Hossain et al. [30] proposed an IDS where the Control Area Network (CAN) Bus communication is the main security concern. The LSTM architecture in their work uses Softmax function and the four attack class types for the proposed IDS are benign, DoS, fuzzing and spoofing. Accuracy, detection rate, recall, F1-score, Area Under the Curve (AUC) as well as Receiver Operating Characteristic Curve (ROC). They use Python and an LSTM model for supervised binary and multiclass classification for NAIST CAN attack dataset achieved a high accuracy of 99.99%.

Proposing techniques that have been improved by the authors is one of the best ways to push the boundary of accurate models and reach even a more ideal approach. Yang and Wang [94] proposed an Improved CNN model that uses the NSL-KDD dataset inside an Ubuntu machine. The implementation of their model used 21 features after the feature selection process from the dataset for four different attacks and evaluation metrics in their proposed IDS approach were accuracy, True Positive Rate (TPR), and FPR. They ultimately acquired 95.36% accuracy using an improved CNN model for both feature extraction and classification.

Davis et al. [71] proposed a hybrid anomaly detection approach for transportation networks that utilizes Extreme Value Theory (EVT) along with the LSTM algorithm. In spite of other investigated approaches, in their approach they did not suffice to a single or two datasets but seven datasets such as Vehicular Travel Time, Vehicular Speed, Vehicle Occupancy, NYC Taxi Demand, Bengaluru Taxi Demand, Electrocardiogram, Bitcoin Prices. Although accuracy was not one of their evaluation metrics, their proposed anomaly detection has had the highest prediction and F1-score among almost all datasets.

Kasongo and Sun [50] proposed a wrapper-based feature extraction method since the significance of feature extraction can be seen in many IDS approaches. Their approach was implemented using Feed-Forward Deep Neural Network (FFDNN). The used datasets were UNSW-NB15-TES and AWID-Min-Tst. 3 hidden layers were used in their approach for both binary and multiclass classification in both datasets and 22 features of the UNSW-NB15-TES dataset and 26 of AWID-Min-Tst dataset were used in Python as the preferred programming language and TPR, FPR, as well as CPU time-consumption were used as evaluation metrics. Finally, their model was compared with algorithms such as SVM, Random Forest, K Nearest Neighbor (KNN), and Decision Tree. The final accuracy of the proposed approach was 99.77% and 99.66% for each dataset respectively.

Zhang et al. [96] proposed yet another approach with their own self-generated dataset inside a vehicular network where Gradient Descent with Momentum (GDM) was coupled with Adaptive Gain (AG) as classifiers. Python and Scikit-learn libraries were used along with BUSMASTER to generate the dataset inside CAN. The evaluation metrics were TPR, FPR, and CPU time consumption. Finally the result after using DNN as the feature extractor algorithm was 98% accuracy. It is worth mentioning that their approach for intrusion detection was the only approach among other investigated methods that are capable of real-time intrusion detection that is to detect threats like intrusions simultaneously with real world data as the model is online and ready to function just like an IDS inside a simulation.

The final investigated approach in wireless networks is from Aloqaily et al. [7] whose work consists of a Decision Tree classifier that takes extracted features from DBN as input. Their approach was implemented using MATLAB and NS-3 as the simulator which yields a 99.43% accuracy with accuracy, detection rate, FPR, FNR, and service retrieval delay being the evaluation metrics of their model. For future work, they have an interest in power transfer throughout the network so the entire network can guarantee the availability of power for every vehicle.

4.2. **Deep learning-based IDS for SDN Environment.** In this section we will discuss all the investigated methods that proposed a deep learning-based IDS approach in the SDN environment. All the details of these approaches are in Table 2. As it can be seen, the first three columns respectively show the name of the authors, the tools they used and whether their approach is a real-time implementation of IDS. The next three columns represent the classification algorithm, the dataset they used, and the feature engineering approach they used. And the final two columns illustrate what evaluation metrics the authors have used as well as the accuracy of their approach.

Novaes et al. [70] proposed an IDS approach that utilizes the GAN algorithm as a classifier for CICDDoS 2019 dataset in the SDN environment. Their method was compared with other classifiers such as CNN, LSTM, and MLP and showed a significant increase when it came to their chosen evaluation metrics such as detection rate, precision, and F1-score. They first discuss the SDN architecture since it is prone to DDoS attacks and implement their approach under such an attack using Floodlight SDN controller, which finally yields a 99.78% accuracy in a real-time environment using only manual feature encoding techniques.

Table 1. Properties of deep learning-based IDS for wireless Environment.

IDS Approach	Simulation Environment	Real Time	Classifier	Dataset	Feature Engineering Method	Evaluation Metrics	Accuracy (%)
Kasongo et al. (2021) [48]	Python, Keras, Tensorflow,	No	Deep GRU	NSL-KDD	Extra Trees Classifier	accuracy, recall, precision, F-Measure	88.42
Riyaz and Ganapathy, (2020) [75]	Python, Tensorflow	No	CNN	KDD-CUP-99	Conditional Random Field-Linear Correlation Coefficient-Based	PRC, ROC, Mean Average Precision	98.88
Kasongo and Sun (2020) [49]	Python, Keras, Tensorflow,	No	Deep LSTM	NSL-KDD	Manual Feature Encoding	accuracy, recall, precision, F-Measure	99.51
Gowdhaman and Dhanapal (2021) [25]	MATLAB R2017b	No	DNN	NSL-KDD	Manual Feature Encoding + Cross-correlation	accuracy, recall, FPR, FNR, precision, F-Measure	95.53
Duan et al. (2020) [15]	Keras	No	CNN	AWID	Manual Feature Encoding	accuracy	99
Sbai and Boukhari (2020) [77]	NS-3	No	DNN	CICDDoS2019	Manual Feature Encoding	accuracy, recall, precision, F-Measure	99.99
Dilipkumar and Durairaj (2021) [14]	MATLAB, NS-2	No	Gradient DBN	Self-Generated	Manual Feature Encoding	DR, Memory consumption, computational time	-
Huang and Lei (2020) [41]	-	No	Imbalance Generative adversarial Network	NSL-KDD, UNSW-NB15, CICIDS2017	Feed-forward Neural Network	accuracy, recall, precision, F-Measure	NSL-KDD = 84.45, UNSW-NB15 = 82.53, CICIDS2017 = 99.79
Hossain et al. (2020) [30]	Python, Keras, Tensorflow	No	LSTM	NAIST CAN	Manual Feature Extraction	accuracy, detection rate, AUC, ROC, F1-scores	99.99
Yang and Wang (2019) [94]	Python, Tensorflow, Ubuntu 16.04	No	Improved CNN	NSL-KDD CUP	CNN	accuracy, TPR, FPR	95.36
Davis et al. (2020) [71]	-	No	EVT + LSTM	Vehicular Travel Time, Vehicular Speed, Vehicle Occupancy, NYC Taxi Demand, Bengaluru Taxi Demand, Electrocardiogram, Bitcoin Prices	-	Precision, Recall, F1-score	-
Kasongo and Sun [50]	Python, Scikit-Learn, Windows 8.1	No	Feed Forward Deep Neural Networks	UNSW-NB15-TES, AWID-Min-Tst	Extra Trees	Accuracy, Precision, Recall	UNSW-NB15-TES = 99.77, AWID-Min-Tst = 99.66
Zhang et al. [96]	Python, Scikit-Learn, BUSMASTER	Yes	GDM + AG	Self-Generated	Deep Neural Network	TPR, FPR, CPU time-consumption	98
Aloqaily et al. [7]	Matlab 2017b, NS-3	No	Decision Tree	NSL-KDD + Self-Generated	DBN	accuracy, DR, FPR, FNR, Service Retrieval Delay	99.43

Abdallah et al. [1] developed an IDS approach that is composed of both LSTM and CNN algorithms. Using CNN algorithm for feature extraction allowed them to format the 48 network features into an image format of 8 x 6 dimensions. They believe that this approach can be utilized as a real time IDS as well in the future. 96.32% accuracy was gained using Python and Keras library in InSDN dataset.

ElSayed et al. [18] performed their IDS on three different datasets, namely CSE-CIC-IDS2018, InSDN, and UNSW-NB15 and the average accuracy for them turned out to be 99.80%, 99.28% and 99.50% respectively. Their approach was compared with other classification algorithms such as LSTM, SVM Logistic Regression (LR), NB, SVM, Random Forest, KNN, and Decision Tree.

The work of Tang et al. [83] proposed an IDS approach that uses deep learning as its basic architecture where two main algorithms, Fully Connected DNN and GRU-RNN, are used as the classifiers of their approach while using manual feature encoding. Their approach yielded a 90% accuracy using POX SDN controller and Python and NSL-KDD for the main dataset.

Another IDS approach that shows promising results in real-time SDN environments is the work of Lee et al. [55] whose approach is denoted as DL-IDPS. Their approach functions in a Ryu SDN controller utilizing Keras and Tensorflow libraries which after creating a self-generated dataset yields a 100% accuracy.

Another real-time IDS approach for DDoS attacks [64] was proposed by Makuvaza et al.. Their approach contains a four-layer DNN for classification and min-max optimization for handling feature encoding. CICIDS-2017 and four of its main features were used in order to train the model and accuracy, precision, recall and F1-score were used as evaluation metrics. The approach acquired 97.59% accuracy which is much higher than the algorithms that the model was compared with, algorithms such as RBM and SVM, GRU and RNN, and finally GRU and LSTM.

Tang et al. [82] presented an IDS approach that uses GRU and RNN coupled with each other. The reason for using RNN algorithm is because it depends on previous computation and its backpropagation helps the model training to be more efficient. The proposed architecture includes three components; flow collector, anomaly detector and anomaly mitigator. They achieved 89% accuracy compared to SVM , DNN, and RNN alone, using the NSL-KDD dataset and POX SDN controller.

Since Openflow protocols are extremely prevalent in SDN environments, Li et al. [59] proposed a real-time defense mechanism against DDoS attacks that take advantage of Openflow protocol. In their architecture, the data is forwarded back and forth between Openflow Switches and the DDoS defender component and consequently flow entries are generated according to the weight of features. The algorithms used in this approach are LSTM, GRU, 3LSTM as well as CNN coupled with LSTM. The achieved accuracy using these algorithms on the ISCX2012 dataset was 99%.

The work of Malik et al. [65] is a hybrid model that takes advantage of both LSTM and CNN together. Their reasoning involves the ability of CNN to extract features and LSTM to prevent the issue of gradient vanishing in RNN. Sigmoid is used as the activation function. Moreover, the model's comparison was with other popular deep learning algorithms such as LSTM coupled with DNN as well as LSTM coupled with GRU. As for the evaluation metrics, accuracy, precision, recall, and F1-score were chosen to finally see that the model yielded a 98.6% accuracy.

Susilo and Sari [80] proposed another IDS approach based on deep learning that uses two classifiers; Random Forest, and CNN algorithms, which is implemented on two distinct datasets, namely BoT-IoT and CIC-IDS-2018. The results show 100% accuracy and 99.95% accuracy for each dataset.

Boukria and Gouerroumi [11] used CICIDS2017 dataset to train their DNN-based algorithm to detect SDN related attacks when it comes to intrusions. The architecture of the control plane of their SDN network has 3 modules; feature extraction modules, feature preprocessing module, and flow classification module. Normalization of data flows includes using a logarithmic function as well as min-max method. Accuracy, precision, recall, and F1-score were used for evaluation metrics and finally the result showed a 99.6% accuracy.

Albahar [3] proposed an approach that can be potentially used in a real-time environment. The approach implements an IDS that runs RNN as the main classifier while a new introduced regularization method is coupled with it, whose functionality is based on standard deviation of the weight matrix. Flow collector, anomaly collector and anomaly mitigator are the main modules that are implemented inside the architecture. Three datasets were used in this approach, which are KDD-CUP-99, NSL-KDD and UNSW-NB15, and the accuracy for each of was 99.5%, 97.39% and 99.9%, respectively.

Choobdar et al. [13] proposed a multiclass approach. Their architecture consists of three phases; firstly, the preprocessing phase that uses Sparse Stacked AE (SSAE) as an unsupervised algorithm extracts the necessary features from data flows. Secondly, the model uses softmax classifier, which is fundamentally part of the CNN algorithm and thirdly, the model optimization takes place. The two datasets used in this work are NSL-KDD and CICIDS2017 and the resulting accuracy for each was 98.5% and 98.9%, respectively.

4.3. **Deep learning-based IDS for IoT Environment.** This section will discuss all the IDS approaches that have been put forward in the IoT environment. All the details of these approaches are in Table 3. As it can be seen, the first three columns respectively show the name of the authors, the tools they used and whether their approach is a real-time implementation of IDS. The next three columns represent the classification algorithm, the dataset they used and the feature engineering approach they used. And the final two columns illustrate what evaluation metrics the authors have used as well as the accuracy of their approach.

Roy and Cheung [76] proposed a novel approach that utilizes bi-directional LSTM RNN, which originates from bi–directional RNN that is capable of processing data both for forward and backward directions. Their approach, which solves the inefficiency of RNN in terms of prolonged time span caused by vanishing gradient, is concerned with detailed features of the dataset during the training phase of the model. Their evaluation metrics show their approach has 95% accuracy using Python and Keras as the simulation tools.

In the work of Dutta et al. [16], we witness the combination of DNN and LSTM that are used for classification. Using DNN, we can see using loss function to penalize the network via back-propagating the errors to adjust the weights. Another interesting collaboration of their work is using Synthetic Minority Over-sampling Technique (SMOTE) followed by the Edited Nearest Neighbors (ENN) to enhance the classification accuracy, which also helps with handling the large scale datasets. By referring to other implementations, they compared their proposed approach with algorithms such as SVM, Random Forest Logistic Regression and MLP, which finally showed their method stands out significantly in all three datasets. Three datasets were used in this work, IoT-23, LITNET-2020 and NetML-2020, and the average accuracy for each was 97%, 100% and 100% respectively.

The focus of AL-Hawawreh et al. [47] is on edge devices within the Brownfield ICS environment. They have implemented a hybrid feature normalization using Denoising AE (DAE) and Sparse Auto-Encoder (SAE) are employed to extract and normalize the features from Gas Pipeline dataset. Precision, FPR and Sensitivity were used as evaluation metrics as well as R programming language to complete the implementation. On the other hand, AL-Hawawreh proposes another IDS approach that is meant for hybrid environments, which we will discuss in the hybrid approaches.

Table 2. Properties of deep learning-based IDS for SDN Environment.

IDS Approach	**Simulation Environment**	**Real Time**	**Classifier**	**Dataset**	**Feature Engineering Method**	**Evaluation Metrics**	**Accuracy (%)**
Novaes et al [70]	Floodlight, Python, TensorFlow, Keras	Yes	GAN	CICDDoS 2019	Manual Feature Encoding	accuracy, Precision, Recall, F1 Score	99.78
Elsayed et al [18]	Python, Keras	No	CNN + LSTM	InSDN	CNN	accuracy, precision, recall, F1-score, AUC	96.32
Tang et al [83]	POX, Python	No	Fully Connected DNN, GRU-RNN	NSL-KDD	Manual Feature Encoding	accuracy, precision, recall, F1-score	90
Lee et al [55]	Ryu, TensorFlow, Keras	Yes	Deep Learning	Self-Generated	Manual Feature Encoding	accuracy, precision, recall, F1-score	100
Makuvaza et al [64]	Keras, Tensorflow, Pandas	Yes	DNN	CICIDS-2017	min-max normalization	accuracy, precision, recall, F1-score	97.59
Tang et al [82]	POX, Python	No	GRU + RNN	NSL-KDD	Manual Feature Encoding	accuracy, precision, recall, F1-score	89
Li et al [59]	Keras	Yes	LSTM, CNN + LSTM, GRU, 3LSTM	ISCX2012	Manual Feature Encoding	accuracy, precision, F1-score	99
Malik et al [65]	POX, Python, TensorFlow, Keras	No	LSTM + CNN	CICIDS2017	Manual Feature Encoding	accuracy, precision, recall, F1-score	98.6
Susilo and Sari [80]	Mininet, Kali Linux, Python, Scikit-learn, Tensorflow	No	Random Forest, CNN	BoT-IoT, CSE-CIC-IDS-2018	Manual Feature Encoding	accuracy, Precision, AUC, ROC	BoT-IoT = 100, CSE-CIC-IDS-2018 = 99.95
Boukria and Guerroumi [11]	Mininet, ONOS	No	DNN	CICIDS2017	Manual Feature Encoding + min-max normalization	accuracy, precision, recall, F1-score	99.6
Albahar [3]	Mininet, POX, Beacon	No	RNN	KDD-CUP-99, NSL-KDD, UNSW-NB15	Manual Feature Encoding + self-introduced regularization	AUC, ROC, TPR, TNR, FPR, accuracy, precision ,F1-score	KDD-CUP-99 = 99.50, NSL-KDD = 97.39 , UNSW-NB15 = 99.9
Choobdar [13]	Mininet, Keras, Tensorflow	No	CNN	NSL-KDD, CICIDS2017	Auto-Encoders	accuracy, precision, recall, F1-score	NSL-KDD = 98.5, CICIDS2017 = 98.8

Kan et al. [84] proposed an approach for IDS that is denoted as Adaptive Particle Swarm Optimization Convolutional Neural Network (APSO-CNN). They have used an evolutionary algorithm called Adaptive Particle Swarm Optimization (APSO) along with CNN, which helped the model to acquire the global fitness value and local optimal fitness value via using the speed of motion for each particle faster. As a result, the APSO-CNN evaluates the cross-entropy loss function and saves the historical information and updates the weights in each iteration.

Telikani and Gandomi [6] propose an IDS using cost-sensitive stacked autoencoder denoted as CSSAE. Their work consists of a couple of steps that innovates the ability of sensitivity towards misclassification of minority classes. Firstly, to reduce the inefficiency of an unbalanced dataset, the computation of sample distributions for cost matrix generation is carried out. Afterwards, SSAE is used to take data as input and finally the training procedure stops one the loss of validation set remains almost the same. The two used datasets were KDD-CUP-99 and NSL-KDD and the accuracy for each was 99.35% and 99%, respectively.

Almiani et al. [69] propose a model that contains two engines, namely traffic processing and deep RNN classification engine. Methods such as min-max normalization as well as manual feature reduction are implemented to make sure of the quality of the dataset, which in this case is NSL-KDD. And as for the Deep RNN classifier, after tuning the model, the results showed a 92.18% accuracy in the end.

Ferdowsi and Saad [19] proposed a real-time IDS for IoT networks that is based on the GAN algorithm. The architecture of the model includes a generator that updates its weights based on gradient descent approach. The model converges to the distribution of the total dataset after a certain number of epochs. Accuracy, Precision, and False Positive were the main evaluation metrics in the model and the final accuracy of the model shows 20% higher accuracy than an average GAN-based IDS.

The approach of Zhang et al. [95] introduces the integration of Genetics Algorithm (GA) and DBN. GA of their approach involves encoding the number of nodes in the three hidden layers directly in the binary chromosome. Roulette wheel selection is implemented in the model and the authors' improvement of the model involves selecting individuals with the greatest fitness value in the GA algorithm. After the feature encoding and dimensionality reduction, RBMs were used within the DBN for the unsupervised learning part of the model. The final results implemented in MATLAB on a self-generated dataset in their approach shows 99.45% accuracy.

The work of Elsaeidy et al. [17] uses deep RBM as the feature classification method since their self-generated dataset contained numerous features. Feed-Forward Neural Network (FFNN), Automated Feed-Forward Neural Network (AFNN), Random Forest and SVM were used as the classifiers. Their main focus on the RBM feature extractor was on high-level features, which was gathered from traffic flow of sensors in the network. Moreover, by stacking RBM on top of each other, they created a deep RBM to outperform compared to other models.

Derhab et al. [89] propose a new IDS based on CNN, which is denoted as Temporal CNN. Their method combines causal padding with CNN to make sure the temporal order is not violated. Their architecture consists of numerous phases including data balancing and feature engineering, training and optimization as well as the final classification. The approach is implemented in Python on the Bot-IoT dataset, which yields 99.99% accuracy.

Gassais et al. [22] use multiple classifiers to detect the anomalies. The authors propose an architecture whose infrastructure relies on sensors and actuators and an analysis system, which is responsible for aggregating the collected data throughout the entire infrastructure. The classifiers are STM, MLP, SVM, Gradient Boosted Trees(GBT), Random Forest and Decision Tree. The final accuracy of the proposed approach shows 100% accuracy in classification of intrusions.

Meidan et al. [66] proposed a new approach using the deep AE for classification. Interestingly, their approach mainly focuses on generating a new dataset for the IoT environment. Their dataset is one the most popular datasets, which we discussed in the dataset section. TPR, FPR and detection time were used as the main evaluation metrics of the approach, which finally we can see a 100% TPR.

Li et al. [60] propose a multi-CNN approach for NIDSs that uses NSL-KDD as the dataset and Tensorflow and Keras as the tools for implementing the experimentation. The multi-CNN model enables the CNN algorithm to exploit the two-dimensional layout of the input data. Their proposed model includes pre-processing modules for data normalization as well as data clustering and model training, which finally allows the model to yield 86.95% accuracy.

4.4. **Deep learning-based IDS for ICS Environment.** In this section, we will take a look at all the approaches that were proposed for ICS environments to detect intrusions. As Table 4 shows, the first three columns respectively show the name of the authors, the tools they used and whether their approach is a real-time implementation of IDS. The next three columns represent the

Table 3. Properties of deep learning-based IDS for IoT Environment.

IDS Approach	Simulation Environment	Real Time	Classifier	Dataset	Feature Engineering Method	Evaluation Metrics	Accuracy (%)
Roy and Cheung [76]	Python (Spyder, Tensorflow)	No	Bi-Directional LSTM RNN	UNSW-NB15	Manual Feature Encoding	accuracy, error rate, precision, FPR,TPR, recall, F1-score	95
Dutta et al. [16]	Python (TensorFlow, Keras ,Scikit-learn)	No	DNN + LSTM	IoT-23, LITNET-2020, NetML-2020	Deep Sparse Auto-Encoder	accuracy, precision, recall, F1 score, Matthews Correlation Coefficient (MCC)	IoT-23 = 97, LITNET-2020 = 100, NetML-2020 = 100
Kan et al [47]	Keras	No	Adaptive Particle Swarm Optimization Convolutional Neural Network	Danmini Doorbell DDb	CNN	Accuracy, precision, Kappa coefficient, Hamming loss, Jaccard similarity coefficient	95
Telikani and Gandomi [84]	-	No	Cost-Sensitive Stacked Auto-Encoder	KDD-CUP-99, NSL-KDD	Two-layer Stacked Auto-Encoder	accuracy, recall, precision, FAR, F-Measure	KDD-CUP-99 = 99.35, NSL-KDD = 99
Almiani et al [6]	MATLAB R2018b	No	Deep Recurrent Neural Network	NSL-KDD	Manual Feature Encoding + min-max normalization	Detection rate, accuracy, precision, F1-score, Matthews Correlation Coefficient (MCC), Cohen's Kappa Coefficient	92.18
Nagisetty and Gupta [69]	Keras	No	SVM, MLP, LR, Auto-Encoder, DNN	UNSW-NB15, NSL-KDD99	Manual Feature Encoding	Accuracy, RMSE, F1-score	99.24
Ferdowsi and Saad [19]	Tensorflow	Yes	Distributed GAN	SBHAR	Manual Feature Encoding	Accuracy, Precision, False Positive	83 (internal attack), 81 (internal attack)
Zhang et al [95]	MATLAB R2016a	No	GA + DBN	NSL-KDD	Min-max normalization	accuracy, detection rate, precision, FAR, recall	99.45
Elsaeidy et al. [17]	MATLAB R2016b, Weka	No	feed-forward neural network (FFNN), Automated feed-forward neural network (AFNN), Random Forest, SVM	Self-Generated	deep RBM	F-Measure	-
Derhab et al. [89]	Python	No	Temporal CNN	Bot-IoT	SMOTE-NC	Accuracy, Precision, Recall, F1-score	99.99
Gassais et al. [22]	Python	Yes	LSTM, MLP, SVM, GBT, Random Forest, Decision Tree	Self-Generated	Manual Feature Encoding	accuracy, Precision, Recall, F1-score	100
Meidan et al [66]	Keras, Wireshark	Yes	Deep Auto-Encoders	N_BaIoT (Self-Generated)	Manual Feature Encoding	TPR, FPR, detection time	-
Li et al [60]	Tensorflow, Keras	No	multi-CNN	NSL-KDD	Manual Feature Encoding	Accuracy, Precision, Recall, FPR, F-score	86.95

classification algorithm, the dataset as well as the feature engineering technique they used. And the final two columns illustrate what evaluation metrics the authors have used as well as the accuracy of their approach.

Wang et al. [90] proposed a deep learning based IDS using Deep reinforcement learning on a Gas Pipeline dataset. The feature extraction method is carried out via utilizing CNN algorithm. The approach also proposes a new feature mapping of data and also a new model of training process. Using Markov decision process and Behrman equation in the deep reinforcement learning the model yields 98.06% accuracy.

Huda et al. [42] propose a Cloud-assisted IoT (CoT) that uses DBN as well as ANN for classification and RBM algorithm for feature encoding. The model is sensitive to malware behavior in the network, all of which are collected and logged inside a Virtual Machine (VM) for feature extraction and finally classification via DBN. The model ultimately yields 99.80% accuracy.

The approach of Lan et al. [53] uses an optimized bidirectional LSTM classifier that works based on the modification of threshold. The authors denoted the approach as Threshold-optimized CNN-BiLSTM-Attention. The modification of threshold involves using ROC curve to reduce the FP and make detection rate more accurate. The model yields 96.7% accuracy in the end.

The work of Liu et al. [62] contributes to ICSs by firstly using CNN for automatic feature extraction. Secondly, their proposed IDS can detect even zero-day attacks as well as increasing the performance and accuracy of an IDS in the ICS environment using Gas Pipeline dataset. The evaluation metrics for providing an assessment of the model were detection rate, FPR and accuracy. For performance comparison, the model is then compared to other traditional methods such as Random Forest, Decision Tree, SVM, Bayesian network (BN) and Hidden Markov Model (HMM).

Alabugin and Sokolov [2] proposed a GAN-based anomaly detection for ICS whose core architecture used bidirectional GAN (BiGAN). Their work is implemented on Secure Water Treatment Dataset (SWaT) with precision and recall being the evaluation metrics. Tensorflow was used to add an auto-encoder structure to the BiGAN architecture to convert the real data space to hidden variable space. Consequently, the network will be able to distinguish objects as well as hidden vectors.

Wang et al. [91] used a stacked deep learning approach for creating models trained on Power System and Gas Pipeline datasets. Accuracy, precision, recall, specificity and F1-score were used as evaluation metrics in the model. Their model consists of constructing 5 networks that range from a relatively small number of neurons to a quite large network with a relatively high number of neurons. The final results for Power System and Gas Pipeline datasets show 98.5% and 99.9% accuracy respectively.

Another approach that focuses on cyber-physical systems [26], which is a type of industrial network, is the work of Li et al. [58]. Their system model, which is demoted as DeepFed, includes trust authority, cloud server, and industrial agents. For the intrusion detection part of their model, they used MLP as well as Softmax as the classifier and CNN coupled with GRU for feature extraction purposes on a self-generated dataset, which finally achieves 99.20% accuracy.

Yang et al. [93] proposed intrusion detection for SCADA systems. In their architecture, once the raw data is retrieved and feature encoding as well as feature extraction on a self-generated dataset is completed using CNN algorithm, the model uses DNN for classification, which yields 99.84% accuracy. The authors proposed a MORE comprehensive approach to cover more attacks in the network in the future.

Süzen's work [81] includes a stacked RBM that ultimately creates a DBN model as the classifier. It is worth mentioning that RBM is used to extract features from a self-generated dataset and afterward encode the features. The approach also uses a Softmax function in addition to accuracy, F1-score, and ROC as evaluation metrics, which yielded 99.72% accuracy in the end.

Chu et al. [21] proposed an approach based on GoogLeNet-LSTM, which is utilized in lieu of conventional CNN algorithm in order to deal with large datasets that have a high number of features. Their approach has various kinds of kernels in a single layer, whereas CNN only relies on one single kernel in a single layer. LSTM coupled with GoogLeNet acquired 97.56% accuracy on the SWaT dataset in the end by using Python and Tensorflow.

Xingjie et al. [92] proposed a method that combines attack trees and LSTM together, in other words, this structure utilizes a tree-like structure to illustrate the relationship and dependency between each step and attack steps. The tree attack events in the attack trees are OR, AND and SAND trees. Once again, the dataset used in another ICS related approach is SWaT dataset, which in the end achieves a 95.4% accuracy.

Li et al. [57] proposed an IDS based on Extreme Learning Machine (ELM) [43]. The authors proposes this model based on the integration of ELM and Sparse AE (AE-ELM), which takes the data from Gas Pipeline dataset and then features are extracted via using SAE, and in the end using MATLAB, ELM is implemented as the classifier so that an accuracy of 97.4% is achieved.

The model's accuracy is shown to be higher compared to other algorithms such as SVM, K-Means, and ELM alone.

Wang, et al. [12] proposed an approach consisting of three parts, ANN network, network training and openmax layer. The ANN is functioning in the model as the main classifier. The network training section is focused on the center loss function, which interacts with the ANN classifier in order to learn from both discriminative and non-discriminative data. And finally the openmax function, which is more comprehensive than softmax function, acs as a detector of unknown attacks in the system. The final results are tested on two datasets, namely, NF-BoT-IoT-v2, Gas Pipeline, which demonstrated highest precision compared to other algorithms.

The approach of Sokolov et al. [79] focused on the Gas Pipeline dataset. Their approach uses two classifiers, one of which comes from RNN algorithm, and that is LSTM, and the other one is GRU algorithm. It is worth mentioning that these two algorithms are not integrated but used separately, which finally yields 91.70% as the highest accuracy using Python and Keras together.

4.5. **Hybrid Approaches.** The investigated approaches were sometimes implemented in a hybrid environment where a single model could perform in a network environment whose identity and network protocols were intertwined and one could not simply distinguish these networks as a single network environment. As Table 5 shows, the first three columns respectively show the name of the authors in hybrid fields, the tools they used and whether their approach is a real-time implementation of IDS. The next three columns show the classification algorithm, the dataset as well as the feature engineering technique they used. And the final two columns illustrate what evaluation metrics the authors have used as well as the accuracy of their approach. This section covers the investigated approaches in regards to hybrid environments where deep learning models were utilized. The investigated approaches in this section were implemented in the following environments:

- SDN + IoT
- SDN + WIreless
- IoT + Wireless
- IoT + ICS

Shu et al. [78] used Python and Tensorflow for their simulation purposes and more importantly they have been able to propose a Collaborative Intrusion Detection System (CIDS) in a VANET network that utilizes SDN protocols simultaneously. Vehicles, Roadside Control Units (RSUs) as well as SDN controllers and cloud servers are the main components of their architecture. Using

Table 4. Deep learning-based IDS for ICS Environment.

IDS Approach	**Simulation Environment**	**Real Time**	**Classifier**	**Dataset**	**Feature Engineering Method**	**Evaluation Metrics**	**Accuracy (%)**
Wang et al. [90]	-	No	Deep Reinforcement Learning	Gas Pipeline	CNN	accuracy, precision, recall, F1-score	98.06
Huda et al. [42]	-	No	DBN + ANN	Self-Generated	RBM	accuracy, FP, FN	99.8
Lan et al. [53]	Tensorflow, Ubuntu 16.04	No	Threshold-optimized CNN-BiLSTM-Attention	Gas Pipeline Testbed	Manual Feature Encoding + min-max normalization	DR, FPR, accuracy	
Liu et al. [62]	Gas pipeline network system Infrastructure	No	CNN	Gas Pipeline Network System	CNN	accuracy, precision, recall, F1-score	94.9
Alabugin and Sokolov [2]	Tensorflow	No	Bidirectional GAN	Secure Water Treatment (SWaT)	Bidirectional GAN	precision, recall	-
Wang et al [91]	Python, Tensorflow, Keras	No	Stacked Deep Learning	Power System, Gas Pipeline	Random Forest	accuracy, precision, recall, specificity, F1-score	Power System = 98.5, Gas Pipeline = 99.9
Li, et al [58]	Python, Keras, Flask, Ubuntu 18.04.3	No	MLP	Self-Generated	CNN + GRU	accuracy, precision, recall, F1-score	99.20
Yang et al. [93]	-	No	DNN	Self-Generated	CNN	precision, recall	99.84
Süzen [81]	Python	No	DBN	Self-Generated	RBM	accuracy, F1-score, ROC	99.72
Chu, et al. [21]	Python, Tensorflow	Yes	LSTM	Gas Pipeline	GoogLeNet	accuracy, FPR, MR	97.56
Xingjie et al. [92]	-	No	LSTM	SWaT	-	accuracy	95.4
Li et al. [57]	MATLAB	No	Extreme Learning Machine [78]	Gas Pipeline	Sparse Auto-Encoder	accuracy, FPR	97.4
Wang, et al [12]	Python, Keras	No	ANN	NF-BoT-IoT-v2, Gas Pipeline	Manual Feature Encoding	precision, recall, F1-score	-
Sokolov et al. [79]	Python, Keras	No	LSTM, GRU	Gas Pipeline	CNN	accuracy, precision, recall	91.70

multiple controllers creates a collaborative network and hence allows the controller to distribute the data in the cloud servers using global minimum. Their model uses two datasets, KDD-CUP-99 and NSL-KDD, for both of which they have been able to acquire 98.37% and 96.77% accuracy respectively. They used accuracy, recall, precision and F1-score as well as AUC as the evaluation metrics of their approach.

Li et al. proposed an algorithm that is capable of extracting features via using deep migration learning [56]. As a result, necessary features from the

KDD-CUP-99 dataset were extracted using MATLAB to help enable the smart city IoT-based network to be protective against cyber security attacks. Since the FAR is high even among the best IDS approaches, their approach uses detection rate, average cost and more importantly FAR to show how efficient their model can be.

The work Polat et al. [73] has been implemented for an SDN-based VANET system inside of a transportation system. Since SSAE is an unsupervised learning algorithm, in this way it is used to extract features from a high-dimension dataset in their work. In addition to a self-generated dataset, SUMO simulator is used to create a transportation system environment, the result of which was a dataset with 42 features. SVM as well as decision tree and KNN were used for comparison and the final result that was implemented in MATLAB demonstrated a 96.9% accuracy in a four-layer SSAE.

Javeed et al. [45] propose an interesting approach that implements IDS in an SDN-enabled environment while taking resource-constrained IoT devices into consideration. CICDDoS2019 is the only used dataset but the classifiers in the proposed approach are LSTM coupled with GRU as well as LSTM with DNN, both of which use Compute Unified Device Architecture (CUDA). The architecture consists of preprocessing the data coming from IoT devices and handling the data flows in the SDN data and control planes. Using the proposed algorithms inside the SDN controller finally achieves 99.74% accuracy using Python and Keras.

Another hybrid work is from Ullah et al. [10] who propose using SDN-enabled for Fog-to-IoT devices. All the Fog-to-Iot devices are implemented beneath the data plane in order for the SDN network to complete the preprocessing phase. Once the feature extraction with CNN is done on the data, the SDN controller is able to do the classification and perform intrusion detection. Evaluation metrics were accuracy, precision, recall, F1-score and ROC, which led to a 99.92% accuracy.

The work of Nie et al. [9] is another real-time deep learning based IDS for IoT devices in transportation networks. The architecture of their proposed work consists of a deep CNN for classification as well as using CNN for implementing feature extraction on a self-generated dataset, whose attributes are taken from link loads of RSUs in the transportation environment. The model is compared with SVM, shallow neural network and Principal Component Analysis (PCA).

Latif et al. [72] proposed an IDS based on a deep RRN in an industrial IoT (IoT) environment, which is denoted as DRaNN. In this work, they used UNSW-NB15 as the dataset and 41 features of it to improve the efficiency of a deep learning-based IDS compared to other proposed approaches before this work. The accuracy of the model was 99.54% in 100 epochs and the evaluation metrics were accuracy, DR as well as FPR.

Nagisetty and Gupta [54] proposed a framework for deep learning based IDS that is composed of five modules, whose functionalities revolve around tasks such as feature transformation, data normalization, model training and classifications. The proposed approach of their work uses UNSW-NB15 as the datasets as well as SVM, MLP, logistic regression, Auto-Encoder and DNN as the classifiers.

As stated earlier in regards to IoT-based IDS approaches, AL-Hawawreh et al. [8] proposed an IDS using Variational AE (VAE) for IoT environment. The usage of statistical probability Distribution in AEs allow their VAE to inherit the same properties, hence the dimensionality reduction in their approach as well as handling the lack of sufficient trained observations is dealt with using VAE. Accuracy, DR and FPR are used as evaluation metrics, which finally gives an accuracy of 92.81%.

5. Conclusion

In this paper, we tried to cover deep learning models and provided a taxonomy of deep learning and non-deep learning algorithms to distinguish the difference between the two. Numerous IDS approaches that used deep learning algorithms were investigated to deeply understand the IDS phenomenon and also understand the importance of each model architecture to finally have better accuracy and performance.

Also, after discussing all investigated methods, it is extremely important to consider the fact that all of the proposed approaches have used different sets of tools and various simulation environments, which can signify that approaches with less accuracy cannot be considered less important since the infrastructure is completely different from other works. Based on the investigated approaches in each field of cyber security, the conclusion that we can draw is that nearly ideal accuracy is achievable in each field of cyber security using deep learning based IDS. However, each approach is bound to certain advantages and disadvantages. We covered these advantages and disadvantages completely in separate sections for each field of cyber security. Table 1 demonstrates comparison of proposed IDS approaches in the wireless environment, Table 2 shows the comparison of proposed IDS approaches in the SDN environment, Table 3 exhibits the compared approaches in the IoT environment and finally Table 4 shows the compared approaches in the ICS environment. Table 5 also shows the comparison of approaches that were used in a hybrid environment.

Table 5. Deep learning-based IDS for Hybrid Environment.

IDS Approach	**Simulation Environment**	**Real Time**	**Classifier**	**Dataset**	**Feature Engineering Method**	**Evaluation Metrics**	**Accuracy (%)**
AL-Hawawreh [9]	R, Weka	No	DNN	Gas Pipeline	Sparse Auto-Encoder + DAE	Precision, FPR, Sensitivity	-
Nie, et al. [72]	LOIC, Wireshark	Yes	Deep CNN	Self-Generated	CNN	accuracy, recall, , False alarm, precision, F-score	97.60
Latif et al [54]	Python	No	DRaNN	UNSW-NB15	Manual Feature Encoding	accuracy, DR, FPR	99.54
AL-Hawawreh and Sitnikova [8]	Python	Yes	Stacked Variational Auto-Encoder	Ransomware and Goodware	Auto-Encoder	Accuracy, DR, FPR	92.81
Polat et al. [73]	SUMO, MATLAB	No	Stacked Sparse Auto-Encoder + Softmax	Self-Generated	Stacked Sparse Auto-Encoder	accuracy, sensitivity, precision, specificity, F1-score	96.9
Javeed et al. [45]	Python, Keras, Tensorflow	No	LSTM + GRU, LSTM + DNN	CICDDoS2019	-	accuracy, precision, recall, F1-score	99.74
Ullah et al. [10]	Python, Keras, Tensorflow, Windows 10	No	LSTM + CNN	Coburg Intrusion Detection Data Set (CIDDS-001)	CNN	accuracy, precision, recall, F1-score, ROC	99.92
Shu et al. [78]	Python, Tensorflow	No	GAN	KDD-CUP-99, NSL-KDD	-	accuracy, precision, recall, F1-score, AUC	KDD-CUP-99 = 98.37, NSL-KDD = 96.77
Li et al. [56]	Matlab R2010b, Ubuntu, Windows 10	No	-	KDD-CUP-99	Deep Migration Learning	DR, False Alarm Rate, average cost	-

In addition, when compared to previous surveys, our work stands out in terms of perspective since we presented a work that investigates IDS approaches within different fields of cyber security. Not only that, but also providing details of each approach and comparing them with other proposed IDSs helps to provide a better understanding of advantages and disadvantages of each approach.

Finally, based on our knowledge for future work, creating a self-generated dataset to increase the accuracy and considering parallel IDSs in an SDN environment improve IDSs performance. We also believe the future of IDSs will be featuring machine learning and deep learning models. The followings are the possible collaborations that can be foreseen in the near future within the field of cyber security:

- Other fields of cyber security such as cloud computing have potential growth when it comes to deep-learning based IDSs. The future works will probably focus more on this area and even hybrid networks such as cloud and SD-WAN environments will draw more attention to themselves.
- Considering IDSs in an SDN environment, parallel IDSs that can work side by side as a Network Function Virtualization (NFV), especially for load-balancing using SDN protocols is also a very palpable idea in the future.
- More and more datasets that are related to blockchain technology are being employed inside the cyber security environment. It is foreseeable to see in the future that more and more approaches adopt this notion.

References

[1] M. Abdallah, N. A. L. Khac, H. Jahromi, A. D. Jurcut, *A Hybrid CNN-LSTM Based Approach for Anomaly Detection Systems in SDNs*, The 16th International Conference on Availability, Reliability and Security (ARES 2021), 34, (2021), 1-7.

[2] S. K. Alabugin, A. N. Sokolov, *Applying of Generative Adversarial Networks for Anomaly Detection in Industrial Control Systems*, 2020 Global Smart Industry Conference (GloSIC), (2020), 199-203.

[3] L. H. Albahar, M. Al, *Recurrent Neural Network Model Based on a New Regularization Technique for Real-Time Intrusion Detection in SDN Environments*, Security and Communication Network, (2019).

[4] A. Aldweesh, A. Derhab, A. Z. Emam, *Deep Learning Approaches for Anomaly-Based Intrusion Detection Systems: A Survey, Taxonomy, and Open Issues*, Knowledge-Based Systems, 189, (2020), 105124.

[5] A. M. Aleesa, B. B. Zaidan, A. A. Zaidan, N. M. Sahar, *Review of Intrusion Detection Systems Based on Deep Learning Techniques: Coherent Taxonomy, Challenges, Motivations, Recommendations*, Substantial Analysis and Future Directions. Neural Comput & Applic 32, (2020), 9827–9858.

[6] M. Almiani, A. AbuGhazleh, A. A.-Rahayfeh, S. Atiewi, A. Razaque, *Deep recurrent neural network for IoT intrusion detection system*, Simulation Modelling Practice and Theory, (2020). 101.

[7] M. Aloqaily, S. Otoum, I. A. Ridhawi, Y. Jararweh, *An Intrusion Detection System for Connected Vehicles in Smart Cities*, Ad Hoc Networks, 90, (2019), 101842.

[8] M. Al-Hawawreh, E. Sitnikova. *Industrial Internet of Things Based Ransomware Detection using Stacked Variational Neural Network*, In Proceedings of the 3rd International Conference on Big Data and Internet of Things (BDIOT 2019), (2019), 126–130.

[9] M. Al-Hawawreh, E. Sitnikova, F. Hartog, *An Efficient Intrusion Detection Model for Edge System in Brownfield Industrial Internet of Things*, Proceedings of the 3rd International Conference on Big Data and Internet of Things (BDIOT 2019). (2019).

[10] M. Arif, I. Ullah, B. A. Raza, A. Sikandar, A. Irshad, S. Baseer, A. Irshad, *Software Defined Network Enabled Fog-to-Things Hybrid Deep Learning Driven Cyber Threat Detection System*, (2021), 6136670.

[11] S. BOUKRIA and M. GUERROUMI, *Intrusion detection system for SDN network using deep learning approach*, 2019 International Conference on Theoretical and Applicative Aspects of Computer Science (ICTAACS), (2019).

[12] C.-H. Chen, C. Wang, B. Wang, Y. Sun, Y. Wei, K. Z. Wang, L. H. Hui, *Intrusion Detection for Industrial Control Systems Based on Open Set Artificial Neural Network*, (2021).

[13] P. Choobdar, M. Naderan, M. Naderan, *Detection and Multi-Class Classification of Intrusion in Software Defined Networks Using Stacked Auto-Encoders and CICIDS2017 Dataset.* Wireless Pers Commun 123, (2022), 437–471.

[14] S. Dilipkumar, M. Durairaj, *Epilson Swarm Optimized Cluster Gradient and Deep Belief Classifier for Multi-Attack Intrusion Detection in MANET*, J Ambient Intell Human Comput, (2021).

[15] Q. Duan, X. Wei, J. Fan, L. Yu, Y. Hu, *CNN-based Intrusion Classification for IEEE 802.11 Wireless Networks*, 2020 IEEE 6th International Conference on Computer and Communications (ICCC), (2020), 830-833.

[16] V. Dutta, M. Choraś, M. Pawlicki, R. Kozik, *A Deep Learning Ensemble for Network Anomaly and Cyber-Attack Detection*, Sensors, (2020), 20, 16.

[17] A. Elsaeidy, K. S. Munasinghe, D. Sharma, A. Jamalipour, *Intrusion detection in smart cities using Restricted Boltzmann Machines*, Journal of Network and Computer Applications, (2019), 135.

[18] M. S. ElSayed, N. Le-Khac, M. A. Albahar, A. Jurcut, *A Novel Hybrid Model for Intrusion Detection Systems in SDNs Based on CNN and a New Regularization Technique*, Journal of Network and Computer Applications, 191, (2021), 103160.

[19] A. Ferdowsi and W. Saad, *Generative Adversarial Networks for Distributed Intrusion Detection in the Internet of Things*, 2019 IEEE Global Communications Conference (GLOBECOM), (2019).

[20] M. A. Ferrag, L. Maglaras, S. Moschoyiannis, H. Janicke, *Deep Learning for Cyber Security Intrusion Detection: Approaches, Datasets, and Comparative Study*, Journal of Information Security and Applications, 50, (2020), 102419.

[21] C. Galdi, A. Chu, Y. Lai, J. Liu, *Industrial Control Intrusion Detection Approach Based on Multiclassification GoogLeNet-LSTM Model*, (2019).

[22] R. Gassais, N. E.-Jivan, J.M. Fernandez, et al., *Multi-level host-based intrusion detection system for Internet of things.* J Cloud Comp 9, (2020), 62.

[23] L. Gonog and Y. Zhou, *A Review: Generative Adversarial Networks*, 2019 14th IEEE Conference on Industrial Electronics and Applications (ICIEA), (2019), 505-510.

[24] P. G. Govind, M. Kulariya, *A Framework for Fast and Efficient Cyber Security Network Intrusion Detection Using Apache Spark*, Procedia Computer Science, 93, (2016), 824-831.

[25] V. Gowdhaman, R. Dhanapal, *An Intrusion Detection System for Wireless Sensor Networks Using Deep Neural Network*, Soft Comput (2021).

[26] S. Han, M. Xie, H. -H. Chen, Y. Ling, *Intrusion Detection in Cyber-Physical Systems: Techniques and Challenges*, IEEE Systems Journal, 8, 4, (2014), 1052-1062.

[27] G.E. Hinton, *Deep Belief Networks*, Scholarpedia, 4, (2009), 5947.

[28] S. Hochreiter, J. Schmidhuber, *Long Short-Term Memory*, Neural Computation, 9, 8, (1997), 1735-1780.

[29] M. Hoque, M. Mukit, A. Bikas, *An Implementation of Intrusion Detection System Using Genetic Algorithm*, International Journal of Network Security & Its Applications, 4, 2, (2012), 109-120.

[30] M. D. Hossain, H. Inoue, H. Ochiai, D. Fall, Y. Kadobayashi, *LSTM-Based Intrusion Detection System for In-Vehicle Can Bus Communications*, in IEEE Access, 8, (2020), 185489-185502.

[31] https://archive.ics.uci.edu/ml/datasets/detection_of_IoT_botnet_attacks_N_Bait

[32] https://kdd.ics.uci.edu/databases/kddcup99/task.html

[33] https://ieeexplore.ieee.org/document/9187858

[34] https://ieee-dataport.org/documents/bot-iot-dataset

[35] https://www.semanticscholar.org/paper/Industrial-Control-System-Simulation-and-Data-for-Morris-Thornton/bb9714e0c661576f5df19fb54e0e26567ca37372

[36] https://www.stratosphereips.org/datasets-iot23#: :text=IoT%2D23%20is%20a%20new,of%20 %20Software%2C%20Prague. Things%20(IoT)%20devices.&text=Its%20goal%20is%20to%20 offer,funded%20by%20Avast

[37] https://www.unb.ca/cic/datasets/ddos-2019.html#: :text=2.,%2Dworld%20data%20(PCAPs).

[38] https://www.unb.ca/cic/datasets/ids-2017.html

[39] https://www.unb.ca/cic/datasets/ids-2018.html

[40] https://www.unb.ca/cic/datasets/nsl.html

[41] S. Huang, K. Lei, *IGAN-IDS: An Imbalanced Generative Adversarial Network Towards Intrusion Detection System in Ad-Hoc Networks*, Ad Hoc Networks, 105, (2020), 102177.

[42] S. Huda, S. Miah, J. Yearwood, S. Alyahya, H. Al-Dossari, R. Doss, *A Malicious Threat Detection Model for Cloud Assisted Internet of Things (CoT) Based Industrial Control System (ICS) Networks Using Deep Belief Network*, Journal of Parallel and Distributed Computing, (2018), 120.

[43] G. B. Huang, Q. Y. Zhu, C. K. Siew, *Extreme Learning Machine: Theory and Applications*, Neu-rocomputing, 70(1-3), (2006), 489-501.

[44] J. Jang-Jaccard, S. Nepal, *A Survey of Emerging Threats in Cybersecurity*, Journal of Computer and System Sciences, 80, 5, (2014), 973-993.

[45] D. Javeed, T. Gao, M. T. Khan, *SDN-Enabled Hybrid DL-Driven Framework for the Detection of Emerging Cyber Threats in IoT*, 10, 8, (2021), 918.

[46] S. Jin, J. -G. Chung, Y. Xu, *Signature-Based Intrusion Detection System (IDS) for In-Vehicle CAN Bus Network*, 2021 IEEE International Symposium on Circuits and Systems (ISCAS), 2021, 1-5.

[47] X. Kan, Y. Fan, Z. Fang, L. Cao, N. N. Xiong, D. Yang, X. Li, *A novel IoT network intrusion detection approach based on Adaptive Particle Swarm Optimization Convolutional Neural Network*, Information Sciences, (2021), 568.

[48] S. M. Kasongo, Y. Sun, *A Deep Gated Recurrent Unit Based Model for Wireless Intrusion Detection System*, ICT Express, 7, 1, (2021), 81-87.

[49] S. M. Kasongo, Y. Sun, *A Deep Long Short-Term Memory Based Classifier for Wireless Intrusion Detection System*, ICT Express, 6, 2, (2020), 98-103.

[50] S. M. Kasongo, Y. Sun, *A Deep Learning Method with Wrapper Based Feature Extraction for Wireless Intrusion Detection System*, Computers & Security, 92, (2020), 101752.

[51] O. Kaynar, A. G. Yüksek, Y. Görmez and Y. E. Işik, *Intrusion Detection with Autoencoder Based Deep Learning Machine*, 2017 25th Signal Processing and Communications Applications Conference (SIU), (2017), 1-4.

[52] A. Khraisat, I. Gondal, P. Vamplew, J. Kamruzzaman, *Survey of Intrusion Detection Systems: Techniques, Datasets and Challenges*, Cybersecure 2, 20 (2019).

[53] M. Lan, J. Luo, S. Chai, R. Chai, C. Zhang, B. Zhang, *A Novel Industrial Intrusion Detection Method based on Threshold-optimized CNN-BiLSTM-Attention using ROC Curve*, 2020 39th Chinese Control Conference (CCC), (2020).

[54] S. Latif, Z. Idrees, Z. Zou, J. Ahmad, *DRaNN: A Deep Random Neural Network Model for Intrusion Detection in Industrial IoT*, 2020 International Conference on UK-China Emerging Technologies (UCET), (2020).

[55] T. -H. Lee, L. -H. Chang, C. -W. Syu, *Deep Learning Enabled Intrusion Detection and Prevention System over SDN Networks*, 2020 IEEE International Conference on Communications Workshops (ICC Workshops), (2020), 1-6.

[56] D. Li, L. Deng, M. Lee, H. Wang, *IoT Data Feature Extraction and Intrusion Detection System for Smart Cities Based on Deep Migration Learning*, International Journal of Information Management, (2019), 49

[57] Y. Li, Y. Li, S. Zhang. *Intrusion Detection Algorithm Based on Deep Learning for Industrial Control Networks.* In Proceedings of the 2019 The 2nd International Conference on Robotics, Control and Automation Engineering (RCAE 2019). (2019), 40–44.

[58] B. Li, Y. Wu, J. Song, R. Lu, T. Li, L. Zhao, *DeepFed: Federated Deep Learning for Intrusion Detection in Industrial Cyber–Physical Systems*, IEEE Transactions on Industrial Informatics, 17, 8, (2021), 5615-5624.

[59] C. Li, Y. Wu, X. Yuan, et al., *Detection and defense of DDoS attack–based on deep learning in OpenFlow-based SDN.* Int J Commun Syst. (2018).

[60] Y. Li, Y. Xu, Z. Liu, H. Hou, Y. Zheng, Y. Xin, Y. Zhao, L. Cui, *Robust detection for network intrusion of industrial IoT based on multi-CNN fusion*, Measurement, (2020), 154.

[61] H. Liao, C. Lin, Y. Lin, K. Tung, *Intrusion Detection System: A Comprehensive Review, Journal of Network and Computer Applications*, 36, 1 (2013), 16-24.

[62] J. Liu, L. Yin, Y. Hu, S. Lv, L. Sun, *A Novel Intrusion Detection Algorithm for Industrial Control Systems Based on CNN and Process State Transition*, 2018 IEEE 37th International Performance Computing and Communications Conference (IPCCC), (2018).

[63] H. Ma, *Pattern Recognition Using Boltzmann Machine*, Proceedings IEEE Southeastcon '95. Visualize the Future, (1995), 23-29.

[64] A. Makuvaza, D. S. Jat, A. M. Gamundani, *Deep Neural Network (DNN) Solution for Real-time Detection of Distributed Denial of Service (DDoS) Attacks in Software Defined Networks (SDNs).* SN COMPUT. SCI. 2, (2021), 107.

[65] J. Malik, A. Akhunzada, I. Bibi, M. Imran, A. Musaddiq, S. W. Kim, *Hybrid Deep Learning: An Efficient Reconnaissance and Surveillance Detection Mechanism in SDN*, IEEE Access, (2020), 8, 134695-134706.

[66] Y. Meidan et al., *N-BaIoT—Network-Based Detection of IoT Botnet Attacks Using Deep Autoencoders*, in IEEE Pervasive Computing, 17, 3, (2018), 12-22.

[67] S. Mohammadi, H. Mirvaziri, M. G. Ahsaee, H. Karimipour, *Cyber Intrusion Detection by Combined Feature Selection Algorithm*, Journal of Information Security and Applications, 44, (2019), 80-88.

[68] N. Moustafa, J. Slay, UNSW-NB15: *A Comprehensive Data Set for Network Intrusion Detection Systems (UNSW-NB15 network data set)*, 2015 Military Communications and Information Systems Conference (MilCIS), (2015), 1-6.

[69] A. Nagisetty , G. P. Gupta, *Framework for Detection of Malicious Activities in IoT Networks using Keras Deep Learning Library*, 2019 3rd International Conference on Computing Methodologies and Communication (ICCMC), (2019).

[70] M. P. Novaes, L. F. Carvalho, J. Lloret, M. L. Proença, *Adversarial Deep Learning Approach Detection and Defense Against DDoS Attacks in SDN Environments*, Future Generation Computer Systems, 125, (2021), 156-167.

[71] D. Neema, G. Raina, K. P. Jagannathan, *A Framework for End-to-End Deep Learning-Based Anomaly Detection in Transportation Networks*, Transportation Research Interdisciplinary Perspectives, 5, (2020), 100112.

[72] L. Nie, Z. Ning, X. Wang, X. Hu, J. Cheng, Y. Li, *Data-Driven Intrusion Detection for Intelligent Internet of Vehicles: A Deep Convolutional Neural Network-Based Method*, IEEE Transactions on Network Science and Engineering, 7, 4, (2020), 2219-2230.

[73] H. Polat, M. Turkoglu, O. Polat, *Deep Network Approach with Stacked Sparse Autoencoders in Detection of DDoS Attacks on SDN-based VANET*, IET Communications, 14, (2020), 4089-4100.

[74] N. M. Rezk, M. Purnaprajna, T. Nordström, Z. Ul-Abdin, *Recurrent Neural Networks: An Embedded Computing Perspective*, IEEE Access, 8, (2020), 57967-57996.

[75] B. Riyaz, S. Ganapathy, *A Deep Learning Approach for Effective Intrusion Detection in Wireless Networks Using CNN.* Soft Comput, 24, (2020), 17265–17278.

[76] B. Roy, H. Cheung, *A Deep Learning Approach for Intrusion Detection in Internet of Things using Bi-Directional Long Short-Term Memory Recurrent Neural Network*, 2018 28th International Telecommunication Networks and Applications Conference (ITNAC), 2018.

[77] O. Sbai, M. El-boukhari, *Data Flooding Intrusion Detection System for MANETs Using Deep Learning Approach*, Proceedings of the 13th International Conference on Intelligent Systems: Theories and Applications (SITA'20), 46, (2020), 1–5.

[78] J. Shu, L. Zhou, W. Zhang, X. Du, M. Guizani, *Collaborative Intrusion Detection for VANETs: A Deep Learning-Based Distributed SDN Approach*, in IEEE Transactions on Intelligent Transportation Systems, 22, 7, (2021), 4519-4530.

[79] A. N. Sokolov, S. K. Alabugin, I. A. Pyatnitsky, *Traffic Modeling by Recurrent Neural Networks for Intrusion Detection in Industrial Control Systems*, 2019 International Conference on Industrial Engineering, Applications and Manufacturing (ICIEAM), (2019).

[80] B. Susilo and R. F. Sari, *Intrusion Detection in Software Defined Network Using Deep Learning Approach*, 2021 IEEE 11th Annual Computing and Communication Workshop and Conference (CCWC), (2021), 0807-0812.

[81] A.A. Süzen, *Developing a Multi-level Intrusion Detection System Using Hybrid-DBN.* J Ambient Intell Human Comput 12, (2021), 1913–1923.

[82] T. A. Tang, L. Mhamdi, D. McLernon, S. A. R. Zaidi and M. Ghogho, *Deep Recurrent Neural Network for Intrusion Detection in SDN-based Networks*,(2018) 4th IEEE Conference on Network Softwarization and Workshops (NetSoft), (2018).

[83] T. A. Tang. L. Mhamdi, D. McLernon, S. A. R. Zaidi, M. Ghogho, F. El Moussa, *DeepIDS: Deep Learning Approach for Intrusion Detection in Software Defined Networking.* Electronics, 9, (2020), 1533.

[84] A. Telikani, A. H. Gandomi, *Cost-sensitive stacked auto-encoders for intrusion detection in the Internet of Things, Internet of Things*, (2021), 14.

[85] M. Usama et al., *Unsupervised Machine Learning for Networking: Techniques, Applications and Research Challenges*, IEEE Access, 7, (2019), 65579-65615.

[86] N. T. Van, T. N. Thinh, L. T. Sach, *An Anomaly-Based Network Intrusion Detection System Using Deep learning*, 2017 International Conference on System Science and Engineering (ICSSE), (2017), 210-214.

[87] K. S. Vanitha, S. V. UMA, S. K. Mahidhar, *Distributed Denial of Service: Attack Techniques and Mitigation*, 2017 International Conference on Circuits, Controls, and Communications (CCUBE), (2017), 226-231.

[88] R. Vinayakumar, K. P. Soman, P. Poornachandran, *Applying Convolutional Neural Network for Network Intrusion Detection*, 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI), (2017), 1222-1228.

[89] X. Wang, A. Derhab, A, E. Aldweesh, A. Z. Khan, F. Aslam, *Intrusion Detection System for Internet of Things Based on Temporal Convolution Neural Network and Efficient Feature Engineering*, (2020).

[90] W. Wang, J. Guo, Z. Wang, H. Wang, J. Cheng, C. Wang, M. Yuan, J. Kurths, X. Luo, Y. Gao, *Abnormal Flow Detection in Industrial Control Network Based on Deep Reinforcement Learning*, Applied Mathematics and Computation, (2021), 409.

[91] W. Wang, F. Harrou, B. Bouyeddou et al. *A Stacked Deep Learning Approach to Cyber-Attacks Detection in Industrial Systems: Application to Power System and Gas Pipeline Systems*, Cluster Comput 25, (2022), 561–578.

[92] F. Xingjie, W. Guogenp, Z. ShiBIN, ChenHAO, *Industrial Control System Intrusion Detection Model based on LSTM & Attack Tree*, 2020 17th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP), (2020).

[93] H. Yang, L. Cheng, M. C. Chuah, *Deep-Learning-Based Network Intrusion Detection for SCADA Systems*, 2019 IEEE Conference on Communications and Network Security (CNS), (2019).

[94] H. Yang, F. Wang, *Wireless Network Intrusion Detection Based on Improved Convolutional Neural Network*, IEEE Access, 7, (2019), 64366-64374.

[95] Y. Zhang, P. Li and X. Wang, *Intrusion Detection for IoT Based on Improved Genetic Algorithm and Deep Belief Network*, in IEEE Access, (2019), 7, 31711-31722.

[96] J. Zhang, F. Li, H. Zhang, R. Li, Y. Li, *Intrusion Detection System Using Deep Learning for In-Vehicle Security*, Ad Hoc Networks, 95, (2019), 101974.

Rasoul Jafari Gohari
Orcid number: 0000-0002-5341-0025
Department of Computer Science
Shahid Bahonar University of Kerman
Kerman, Iran
Email address: rjafari@math.uk.ac.ir

Laya Aliahmadipour
Orcid number: 0000-0002-0706-5611
Department of Computer Science,
Shahid Bahonar University of Kerman,
Kerman, Iran
Email address: l.aliahmadipour@uk.ac.ir

Marjan Kuchaki Rafsanjani
Orcid number: 0000-0002-3220-4839
Department of Computer Science
Shahid Bahonar University of Kerman
Kerman, Iran
Email address: kuchaki@uk.ac.ir