International Journal of Computers and Informatics

Journal Homepage: https://www.ijci.zu.edu.eg

Int. j. Comp. Info. Vol. 10 (2026) 20–55

Paper Type: Original Article

Deep Learning-Based Intrusion Detection Systems for IoT Networks: A Systematic Literature Review and Comparative Analysis

Salma A. Walli [1,*]

[1] Department of Computer Science, Faculty of Computers and Informatics, Zagazig University, Zagazig 44511, Egypt.
Email: 20912019200851@fci.zu.edu.eg.

Received: 16 Oct 2025 **Revised:** 30 Nov 2025 **Accepted:** 26 Jan 2026 **Published:** 27 Jan 2026

Abstract

The new generations of communication networks are demanding intrusion detection systems capable of addressing sophisticated cyber threats. This systematic literature review examines deep learning-based intrusion detection systems for IoT networks through rigorous analysis, adhering to PRISMA 2020 guidelines. We synthesize findings from studies to address five research questions covering IoT security challenges, architectural approaches, performance characteristics, emerging research directions, and providing taxonomy of deep learning architectures, strategies and applications. Our analysis indicates that hybrid deep learning architectures report higher metrics than single-model approaches in evaluated scenarios. Critical research gaps emerge across multiple dimensions, such as edge deployment limited resources, lack of realistic IoT-specific datasets and absence of explainable AI mechanisms in current solutions. This synthesis provides insights for advancing IoT security.

Keywords: Intrusion Detection Systems, Deep Learning, Internet of Things, Cybersecurity, Neural Networks, Network Security, Systematic Literature Review

1 | Introduction

The rapid expansion of global networks and their associated technological developments led to several internet security issues. Various security threats raised due to uncontrolled access of information posing challenges to network security and intrusion detection effectiveness. To prevent the network from potential intrusions, IDSs arises as a tool to ensure network confidentiality, integrity, and availability by monitoring network traffic [1],[2].

Recently IoT networks has been adapted by several organizations to automate and optimize their business, it is considered as one of the fastest growing technologies. A new security challenges arises due to the continuous advancement of IoT technology. The connection between IoT devises usually takes place through a wireless network, such an environment easily enable attacker to gain illegal access to network devices. Traditional security techniques such as encryption, authentication, access control, network security, and application security were used to address IoT security challenges, however those mechanisms have been proven their inefficiency to meet environments diverse contexts, however targeting a specific security threats by implementing a security measurements against it can be more effective. Recently a new sophisticated attack have been arise, therefor exploring more effective IDSs is the research main goal. IDSs are potential methods

Corresponding Author: 20912019200851@fci.zu.edu.eg

for tracking IoT environments against attacks which present at network level. The IDS analyses the network data packets while offering a real-time responses considering conditions such as low energy, low process capacity, rapid response, and massive amounts of data processing [3],[4]. Most of existed IDSs have various drawbacks including limited flexibility and scalability[5].

One of most common issues that impact ID techniques is availability of labeled datasets, although they are costly, they provide higher accuracy to IDS. Based on the availability of labeled data, the associated learning strategy applied to achieve higher detection accuracy, on the other hand utilizing unsupervised learning methods is necessary when no labeled data is available even if they result in lower accuracy and higher false positives rates. Most of present IDS using various machine learning approaches to detect normal and abnormal traffic in systems and networks, however they still suffer from higher false positive and lower detection rates. To deal with various types of intrusions and security challenges in varied environments, DL have been adapted as a subset of ML. DL methods are capable of modeling complicated data architectures to execute varied non-linear data transformations and recognize patterns in data using several architectures. ANNs are the basic architecture of DL models, they process data using several hidden layers. Various deep learning networks have been deployed in different domains and intrusion detection contexts, such as DBNs, DNNs, CNNs, Generative adversarial networks GANs, and RNNs. Supervised, and semi-supervised and unsupervised learning techniques can be utilized. Furthermore, deep learning promotes incremental learning and can extract novel features from training data samples [6].

2 | Review Objectives and Questions

This review provides a deep understanding of deep learning approaches used in IoT intrusion detection by investigating five different research questions that are interconnected:

RQ1: What security challenges and attack vectors distinguish IoT environments from traditional network infrastructures, and how do these differences shape defensive requirements?

RQ2: Which deep learning architectures have employed for intrusion detection, and what their advantages and limitations?

RQ3: How have researchers adapted deep learning methods for IoT intrusion detection with resource constraints and architectural considerations?

RQ4: How do deep learning-based intrusion detection approaches evaluated under controlled experimental conditions?

RQ5: What critical research gaps, challenges, and open directions exist for advancing deep learning-based intrusion detection?

This review presents the following:

• Surveys deep learning based approaches to detect intrusions in IoT systems by examining various architectures, and performance metrics across fifty peer reviewed studies.

• Provides a comparative analysis of the performance of multiple deep learning architectures.

• Assesses the difficulties associated with of intrusion detection systems in IoT environment.

• Classifies the research gaps in the field of deep learning based IoT intrusion detection.

Section 3 describes the systematic review methodology, Section 4 describes the IoT foundation and reviews relevant survey literature, Section 5 analyzes the deep learning architectures employed for detecting intrusions, Section 6 contains an experimental evaluation of the performance of the deep learning approaches, Section 7combines the results of the study, implementation challenges, and research paths, Section 8 concludes and summarize the review by providing findings and recommendations.

3 | Methodology

We used Preferred Reporting Items for Systematic Reviews and Meta-Analyses PRISMA 2020, which allows to conduct a systematic literature reviews. PRISMA has become an international standard for systematic reviews in all scientific areas and it promotes the quality of reports. The review process includes the formulation of a research question, the systematic search for studies, and the selection of studies according to determined procedure.

3.1 | Review Protocol and Questions

A systematic review protocol created before literature search to clearly define the aims of the review and include details on specific inclusion and exclusion criteria, as well as search methods and data extraction techniques.

3.2 | Information Sources and Search Strategy

Search strategy was implemented across multiple academic databases and digital libraries to identify relevant peer-reviewed publications. Information sources included:

- IEEE Xplore Digital Library
- ACM Digital Library
- Elsevier ScienceDirect
- Springer Link
- MDPI Publishing Platform
- Wiley Online Library
- Google Scholar

We include publications that focused on recent advances in deep learning-based intrusion detection. The search strategy uses combination of keywords organized using Boolean operators:

- Intrusion Detection: ("Intrusion detection" OR "anomaly detection" OR "network security" OR "cyber threat detection" OR "attack detection")
- Deep Learning: ("deep learning" OR "neural network" OR "convolutional neural network" OR "CNN" OR "recurrent neural network" OR "RNN" OR "LSTM" OR "autoencoder" OR "deep neural network" OR "DNN")
- IoT: ("Internet of Things" OR "IoT" OR "Industrial IoT" OR "IIoT" OR "smart devices" OR "edge computing" OR "IoT networks")
- The complete search string combined these clusters as follows:
    - (Cluster 1) AND (Cluster 2) AND [(Cluster 3) OR "systematic review" OR "literature review" OR "survey"

3.3 | Eligibility Criteria

Studies were selected by defined criteria to ensure that studies chosen were relevant to the study objectives.

Inclusion Criteria:

- Peer reviewed journal articles, conference papers, and systematic reviews.
- Deep learning-based studies on intrusion detection, anomaly detection, and other cybersecurity applications
- Studies that addressed IoT security, Network Intrusion Detection, Other areas of Cybersecurity
- Publications that included details on the Deep Learning architecture, data sets used, and evaluation metrics applied in the study
- Publication in high-quality, peer reviewed journals/venues
- Study articles written in English language

Exclusion Criteria:

- Preprint publications, Technical Reports, White Papers, Thesis Documents
- Publications prior to 2020
- Traditional studies which did not include Deep Learning component
- Non-Intrusion Detection Studies, Non-Network Security Studies, Non-Cybersecurity Studies
- Duplicate Publications and Extended Versions of publications

• Predatory Journals/Venues that lack a Rigorous Peer Review Process.

3.4| Study Selection Process

Study selection followed multi-stage screening process:

• Identification: Initial database searches as we aggregated search results from all databases and imported them into reference management software for processing.
• Duplicate Removal
• Title and Abstract Screening
• Full-Text Assessment
• Final Inclusion: Studies satisfying all eligibility criteria after full-text assessment were included in the systematic review

3.5| Data Collection and Extraction

We developed method for extracting data to ensure that a data were extracted from the studies:

Bibliographic Data Elements:

• Authors and publication date
• Title
• Venue/Publisher
• Study Type

Data Elements Describing Methodology:

• Deep Learning Architecture
• Network Layers, Activation Functions, Optimization Algorithms
• Training Procedures, Hyperparameters Configured

Context of Application:

• Target Environment
• Layer of IoT Addressed
• Types of Attacks Detected
• Data Elements Relating to Datasets and Evaluation
• Evaluation Metrics
• Results of Performance and Comparative Analyses

Data Elements Relating to Quality Indicators:

• Experimental Design
• Validation Methods
• Code Availability

4| Foundational Concepts and Background

4.1| Layered Architecture Models

Several IoT architectures are proposed by researches in the literature. Architectures such as the 3-layer architecture which comprise the perception or device, network or transmission and application layers. The ITU recommended Reference Model for IoT which constructed from four layers device layer, network layer, application support layer, service support and application layer. IoT-A Architectural Reference Model proposed by FP7 which is describe the modeling of IoT business process management, IoT services, cross-service organization and virtual entities, information and functional aspects. An IoT Reference Architecture developed by WSO2 that comprise five layers including Client/external communications, Event processing and analytics, Aggregation/bus layer, Relevant transports, Devices. And An IoT Reference Architecture suggested by Cisco which is seven layered IoT reference model [7].

We aligned to Cisco IoT seven-layered reference model which is present in Figure 1 to facilitate more about model layers, protocols, IoT common attacks, and IDSs.

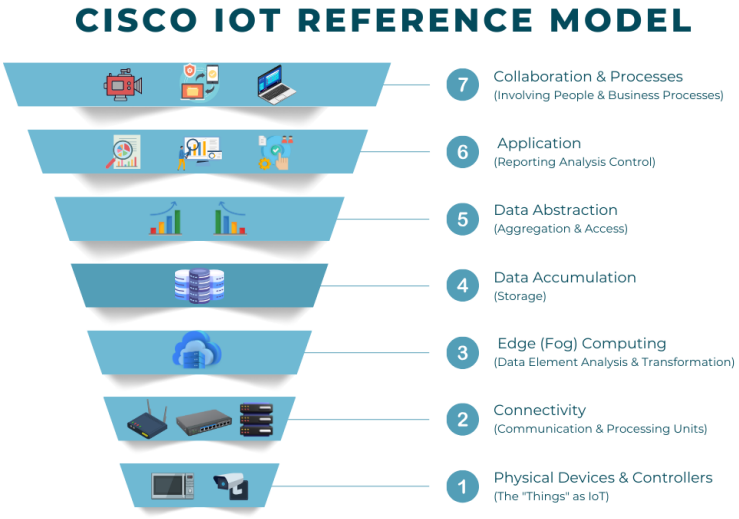


Figure 1 CISCO IoT seven-layer layer reference model.

4.2| Security Challenges in IoT Environments

IoT deployments encounter several security limitations stem from the fundamental differences between IoT and traditional networks. The security issues caused by several characteristics, including [8]:

- Limited Resources, in which IoT devices are constrained by available resources
- Devices diversity, IoT networks consist of several devices and applications
- Poor Security standards
- Changing Network topologies and architectures are constantly.
- Large Attack Surfaces across Physical, Network, and Application Layers.

4.3| Review of Related Research

In this literature, we collect some surveys to provide an overview of IDS in several domains including IoT, CTI mining, communication networks, network traffic analysis and metaverse. Most of the listed surveys and literature reviews focus on IDS in IoT representing the main focus of the literature. This section provide a comparative analysis takes place in Table 1 to compare the existing survey papers based on some criteria encompassing focus area of study, review methodology, existence of experimental study, taxonomy, survey key findings, and future directions. In the following we will discuss the main contribution of the existing surveys:

Nuaimi, Fourati, and Hamed [9] conducted a PRISMA based SLR on the recent ML and DL approaches for IDS in IIoT. The survey proposed a taxonomy classify the collected papers based on six categories including placement strategy, detection method, validation metric, IIoT use case, and ML techniques. As a main finding this research stated that, most of publications relays on centralized placement strategies for intrusion detection, while employing a hybrid strategy integrating centralizes and distributed placement strategies for ID will achieve better performance.

The survey presented by Lee et al. [6] provides a comprehensive analysis on DL- based IDS by building a taxonomy to classify the deep learning-based IDS schemes based on the utilized DL network. The survey also propose prior survey issues in the context of deep learning-based IDS such as limited application of blockchain technology, dependency on general network datasets without paying attention to the new emerging cyberattacks and imbalanced datasets. As a future direction the authors of this work recommends using online and incremental learning for mitigating real-time intrusion detection.

Asharf et al. [7] provides a literature review on ML and DL approaches for IDS in IoT networks, also surveys various dataset related IoT security research. The author extensively discussed the concept of IoT presenting its architecture, protocols, systems vulnerabilities, and protocol-level attacks. This work provide a comprehensive summary for challenges encountered by IoT environments and networks and suggested solutions, highlighting IDS based ML and DL methods.

Wang et al. [10] briefly reviewed IDS and how they can be employed for detecting and classifying cyber-attacks using techniques such as ML, DL, and SWEVO. The survey also investigated IDS taxonomy, feature engineering techniques, IDS based computational intelligent methods, datasets, performance metrics and a wide range of application for IDS. However as the previous work, the author focused on using representative benchmark datasets as it impacts the quality of selected feature and then detection rates, also limitation of each dataset discussed in the scope of the literature. As a future direction the review proposed using a hybrid approach integrating ML and DL methods with SWEVO techniques to enhance intrusion detection efficiency.

The authors of this work Chou and Jiang [11] introduce a taxonomy of data-driven network intrusion detection algorithms based on challenges to research studies and technical methods and analyzing public dataset within the proposed taxonomy. One of the survey findings is that future research should focus on massive network data, streaming and dynamic data, and real-world network data collection and availability. Also the survey concluded that there is a lack of real-world network data, particularly from consumer networks, that can affect the performance of model in simulated network within real-world network traffic data.

Sun et al [12] provides a survey covering cutting-edge CTI mining researches which reveals its ability for improving cyberattacks defense capabilities. This work classifying the existing work in this area of study based on the objective of gathering CTI knowledge emphasizing the used approaches within the prior studies. Based on the proposed classification the survey discusses the existing work focusing on cybersecurity entities and events, cyberattack tactics and processes, hacker profiles, and signs compromise, vulnerability exploits, malware deployment, and threat hunting. Additionally the literature reviewed existing challenges and potential future directions.

Recently, Awadallah et al. [13] provides an overview over cybersecurity attacks related to data, identity, user privacy, digital wellbeing, legal regulations, and NFTs using the metaverse enabling technologies. The survey analyzes several AI techniques to mitigate cyber security in the metaverse, including user authentication, intrusion detection systems, and blockchain security. As a main finding this work incorporates several biometrics along with EEG and ECG to give liveness checks and validation of NFT transactions and users in the metaverse. Moreover investigating several AI approaches for intrusion detection in the metaverse highlighting their importance in securing blockchain and NFT transactions by detecting fraud, ensuring smart contract security, and verifying content.

Kheddar et al. [14] provide an analysis of how RL-based intrusion detection systems automate real-time detection, reduce false positives, and improve capabilities. Based on comparisons with previous surveys the authors extends their research to cover various areas of study including IoT, ICSs, cloud computing, smart grids, and various other domains. The survey proposed a taxonomy facilitating RL and DRL techniques and application particularly IDSs. Also covering important issues include adaptation to dynamic situations,

scalability, interpretability, and robustness against adversarial attacks. Additionally, the survey suggests hybridizing classical IDSs with RL-based IDSs addressing energy efficiency and using LLMs.

Nascita et al. [15] extensively reviewed XAI approaches in NTA including traffic classification and prediction, classifying cyberattacks and intrusion detection, covering methodologies, applications, requirements, issues and future directions. The survey discusses the pivotal role of XAI in improving network security, performance and reliability. Moreover challenges and gaps arises from implementing XAI in NTA.

Lampe and Meng [16] systematically reviewed IDSs in automotive domain ranging from inter-vehicles network to intra-vehicles network facilitating applied methodologies and their performance rates in this domain. The authors of this work comprehensively cover various methodologies including non-learning, traditional ML, DL, and blockchain protected methods, also they provide a comparative analysis to their performance according to several evaluation metrics. The articles also stated open challenges to automotive IDSs such as false positive rate data and training requirements, detection accuracy and latency against consumed resources and critical challenges with their subsequent solutions.

The survey paper provided by Zipperle et al. [17] reviewed PIDS, highlighting the drawbacks of traditional IDS to cope with prevalence of sophisticated attacks due to their high false alarm rates. The survey discusses several issues to PIDS including privacy concerns, run time overhead, scalable graph summarization techniques inadequacy and insufficiency of real-world benchmark dataset. Also the survey proposed future directions focused on maintaining privacy while capturing data provenance, reducing storage overhead using scalable graph summarization methods utilize lossless and lossy reduction techniques and utilizing real-time intrusion detection methods.

The article introduced by Halvorsen et al. [18] provide a mapping study and a review analysis on how GMLMs can be employed to address ML based IDSs issues. Also this literature offer three areas where GMLMs can be applicable to overcome issues with intrusion detection including penetration testing, GMLMs as IDSs and supplementing datasets. The authors of the work successfully proven the effectiveness of GMLMs in detecting attacks that can be ignored by ML based IDSs, moreover introducing new minority classes to the dataset and then enhancing the performance of IDSs which trained on those new datasets. As a key finding the article stated that GMLMs can perform better than other traditional methods for intrusion detection.

Table 1 Comparetive analysis of the existing survey papers

Ref	year	Focus area	Review methodology	Experimental study	Taxonomy	Key findings	Future directions
[9]	2023	IIOT (Perception layer, network layer, data processing layer, and the application layer)	PRISMA	No	No	Promising performance of hybrid IDS placement strategy over other strategies , especially by integrating FL centralized placement strategy with Blockchain distributed placement strategy	Yes
[6]	2021	IOT(data processing layer), SDNs, vehicular networks, and general environments	A survey-based systematic approach	No	Yes	Effectiveness of DL in feature extraction and classification stages of IDS. Highlighting prior researches issues such as: (blockchain limited usage, dependency on	Yes

						general network datasets and misuse of hybrid schemes, etc.)	
[7]	2020	IOT (Perception layer, network layer, and the application layer)	None	No	Yes	Challenges encountered by IOT NIDS such as (features limitations with public dataset, high false alarms rates and complexity of ML and DL NIDS, etc.). Proposing solutions such as (providing high quality datasets and use of ensemble ML and DL over individual ML algorithm, etc.)	Yes
[10]	2022	IOT (Perception layer, network layer, and the application layer)	A survey-based approach	No	Yes	Promising performance of ML, DL, and SWEVO in IDS. Challenges: (High False Alarms and Zero-day Attacks handling, etc.) and future directions such as: (Incorporating representative datasets for efficient feature extraction and enhancing ML and DL methods by hybridizing them with SWEVO, etc.)	Yes
[11]	2022	IOT (Perception layer, network layer, and the application layer)	None	No	Yes	Classifying data-driven network intrusion detection techniques based on challenges to research studies and technical approaches. Areas needed to be addressed more in the future due to lack of studies: (big network data, streaming and changing data, and real-world data collection and availability).	Yes
[12]	2023	Cyber Threat Intelligence Mining IOT (Perception layer, network layer)	CTI mining methodology	No	Yes	Developing CTI mining methodology for proactive cybersecurity protection. Analyzing the state-of-the-art methods for CIT knowledge acquisition taxonomies.	Yes
[13]	2024	Metaverse	A survey-based approach	No	Yes	Presenting different cybersecurity attacks base on features and enabling technologies of the metaverse to identify attack incidents and how they can spread through metaverse. Incorporates several biometrics along with EEG and ECG to give liveness checks and validation of NFT transactions and users in the metaverse.	Yes

						Effectiveness of AI for ID in metaverse, blockchain and NFT transactions security.	
[14]	2024	Communication Networks including IOT as a subset (Perception layer, network layer, and the application layer)	Article collection and selection and Bibliometric analysis	No	Yes	Investigating new areas like the combination of RL-based IDSs with conventional methods, energy efficiency considerations, application of LLMs and hybrid techniques. analyzing existing literature gaps and suggesting future directions	Yes
[15]	2024	Network Traffic Analysis including IOT (Network layer, and the application layer)	Wohlin [19] systematic approach	No	Yes	The Promising performance of XAI into NTA, which improves understanding of network behavior and decision-making. Implementing XAI in this domain posing challenges including (Insufficient methods and integration cost, etc.)	Yes
[16]	2023	Automotive domain : IVNs IoV ITS	None	No	Yes	Systematically reviewing intrusion detection within automotive domain (intra and inter vehicles networks). Highlighting the performance of IDSs which based on non-learning, traditional ML, DL methods accordingly and blockchain protected methods as they applied to federated IDSs.	Yes
[17]	2022	Data Provenance : Information flow among system entities	A survey-based approach	No	Yes	Effectiveness of PIDS to capture highly sophisticated attacks and reducing false-alarm rates excelling traditional IDSs. Analyzing previous surveys in this domain offering gaps and future directions.	Yes
[18]	2024	Penetration testing GMLMs as IDSs Supplement datasets	A survey-based approach	No	Yes	Issues encountered by traditional ML approaches such as data quality and performance can be overcome by GMLMs. Effectiveness of GMLMs with introducing new minority classes to the dataset.	Yes

5| Deep Learning Architectures for Intrusion Detection

Several DL Techniques proposes by researches in this domain for the purpose of enhancing IDSs capabilities, therefore a comprehensive study on DL methods proposed in this literature, Figure 2, Figure 3 present a taxonomy of DL methods based on model architecture and learning strategy, and Figure 4 present common DL application. Also in this section we review selected architectures to our experimental study.

The CNN-based model proposed by Aljuaid and Alshamrani [20] for enhancing IDSs within cloud computing environments. Authors of this wok employed seven stages approach involved data preprocessing, feature selection using Pearson correlation matrix analysis, SMOTE and down-sampling techniques to address data imbalance. The used CNN architecture consists of three Conv1D layers, a batch normalization for reliable training process, MaxPooling1D layers followed by a Flatten layer to converts the feature maps to one-dimensional vector and another two fully connected dense layers. To prevent overfitting a Dropout layer with a 0.5 dropout rate added, and the output layer is dense layer with SoftMax activation function.

The BiDLSTM model which is introduced by Imrana et al. [21] to overcome limitations of traditional LSTM architecture for intrusion detection. The proposed architecture constructed from embedding layer to map inputs to their representations, feeding it to bidirectional LSTM layers, the output then fed to fully connected layers ReLU activation function. Also a dropout layer of 0.2 dropout rate included to ensure that the model doesn't over-fit.

The RNN model proposed by Wang et al. [22] tackling intrusion detection. RNN architecture used in this paper consists of five recurrent layers, five dropout and batch normalization layers featured to prevent overfitting and output layers.

The DLSTM classifier with proposed by Kasongo and Sun [23] to be included within a wireless IDS. The model structured from an input layer, a DLSTM Unit that include LSTM layers with and a Dense Feed Forward Layer with ReLU and Sigmoid activation functions, also an output layer with softmax activation function.

HCRNNIDS introduced by Khan [24] as a hybrid model that combines CNNs and RNNs within IDS to enhances detection capabilities and reduce false alarm rates. Another hybrid model DCNNBiLSTM proposed by Hnamte and Hussain [25] that model combines CNN and BiLSTM layers, followed by a DNN layer for IDS.

CNN–WDLSTM which proposed by Hassan et al. [26] is a hybrid deep learning approach that integrates CNNs with WDLSTM to enhance intrusion detection within big data environments. The proposed model comprise two 1D convolutional layers with ReLU activation function, one 1D maximum-pooling layer, one 1D WDLSTM layer, and one fully connected layer.

The GRU based DL architecture also introduced by Ansari et al [27] for predicting network intrusion alerts by learning from historical alerts generated by malicious sources. The model contracted from three-layer GRU layers stacked on the top of a dense layer. Also IDSs DL based architectures Learn compact representations of data by using a encoder-decoder, which able to Learn to detect anomalies based on the difference between what is expected versus what was actually received. This method has advantages in cases where labeled attack data is rare or unavailable, also used for detecting new sophisticated attacks [6],[28].

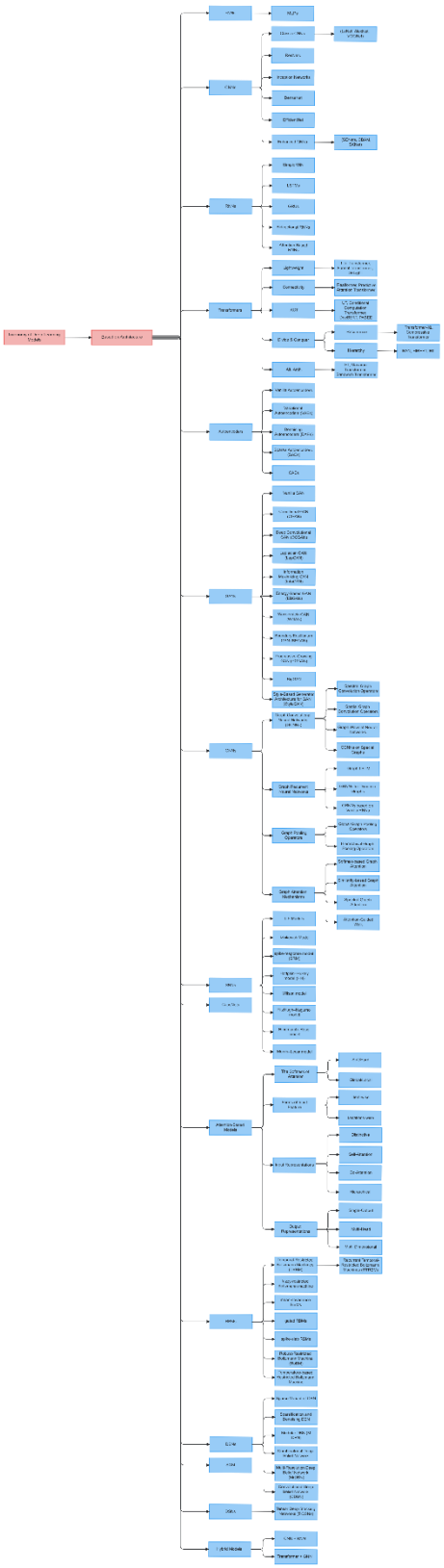


Figure 2 a Taxonomy of Deep Learning Models based on architecture

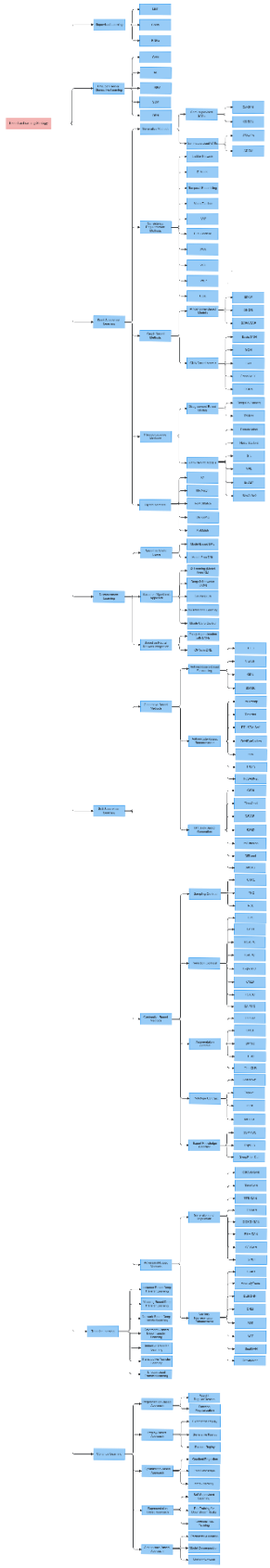


Figure 2 A Taxonomy of Deep Learning Strategies

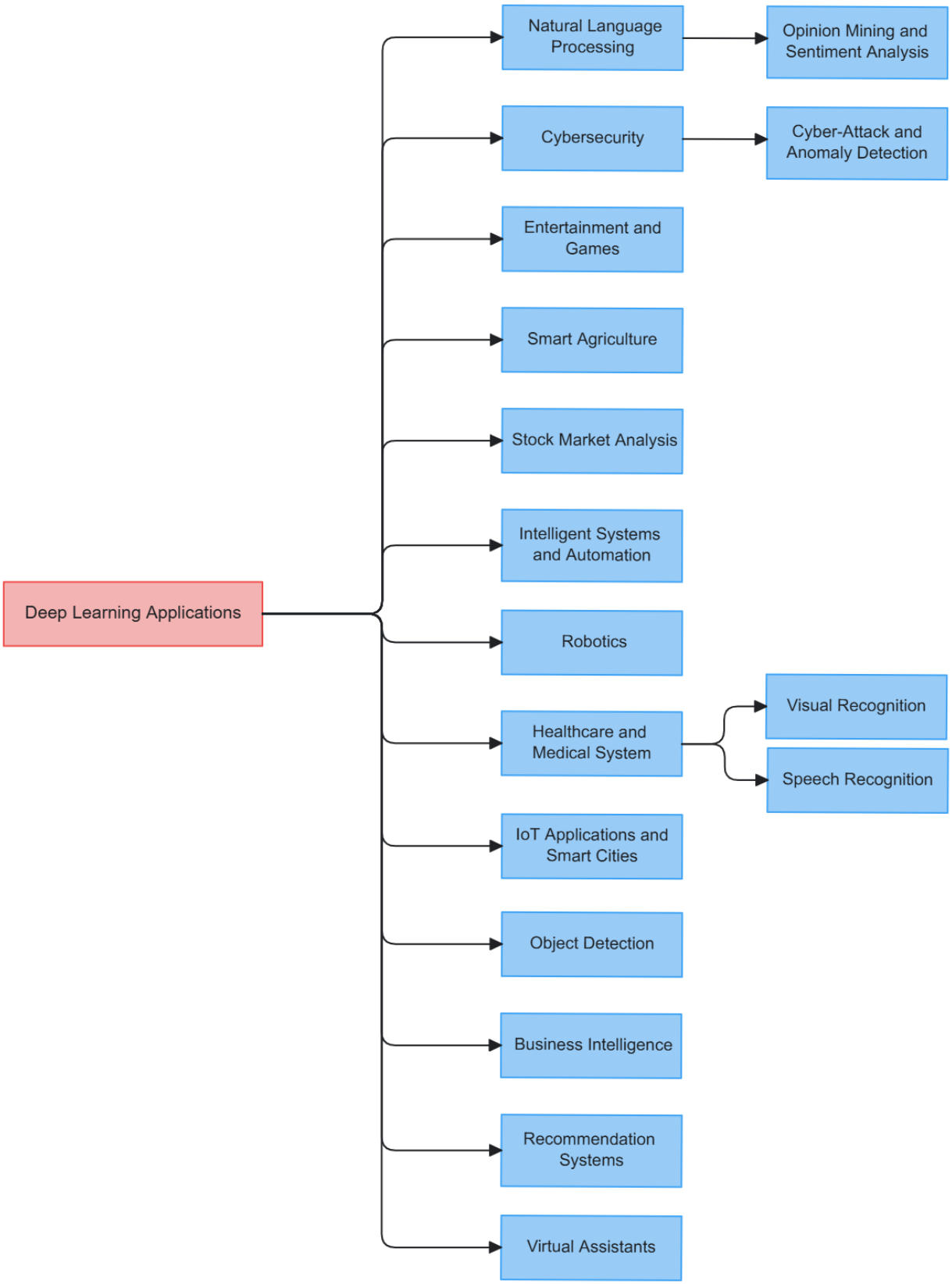


Figure 3 A Taxonomy of Deep Learning Applications

6| Experimental Analysis and Performance Evaluation

In this section we present a systematic analysis of experimental studies, evaluating deep learning-based intrusion detection systems. We introduce dataset characteristics, evaluation methodologies, performance metrics, and comparative results.

6.1| Benchmark Datasets for IDS Evaluation

We have analyzed and compared a number of deep learning-based intrusion detection systems using benchmark datasets including Edge-IIoTset, IEC 60870-5-104, and DNP3 Intrusion Detection datasets, which contain various attacks, traffic types and operating conditions.

Edge-IIoTset was published by [29] as inclusive realistic cybersecurity dataset collected under seven-layers IoT testbed configured with demonstrative set of devices, protocols and cloud and edge setups. In this testbed, the IoT traffics were generated from more than 10 IoT devices including ultrasonic sensor, flame sensor, affordable digital sensors for measuring humidity and temperature, pH sensor meter, Soil moisture sensor, water level sensor, soil sensor, etc. The data included fourteen categories of attacks associated with IoT connectivity protocols. These attacks were belonged to five threats namely Man in the middle attacks, Malware attacks, Information gathering (IG) attacks, Injection attacks, and DoS/DDoS attacks. It also includes normal traffic samples. The Edge-IIoTset dataset contains a set of 61 highly correlated features from a total of 1176 features, which are aggregated from several sources such as system alerts, logs, network traffic, etc. The class distribution of samples across both training and test is introduced in Table 2.

Table 2 Summary of class distribution of the Edge-IIoTset data.

Attack family	Attacks class	Training samples	Testing samples	Class weights
DoS/DDoS	TCP SYN Flood DDoS	8198	2049	2.9553
	UDP flood DDoS	11598	2900	1.2170
	HTTP flood DDoS	8396	2099	2.9642
	ICMP flood DDoS	10477	2619	1.2706
IG	Port Scanning	7137	1784	6.5568
	OS Fingerprinting	682	171	147.7989
	Vulnerability scanning attack	8050	2012	2.9524
Injection	XSS	7634	1909	9.2961
	SQL injection	8225	2057	2.8894
	Uploading attack	8171	2043	3.9312
MiTM	ARP Spoofing	286	72	121.8672
	DNS Spoofing			
Malware	backdoor	7892	1973	5.9507
	Password cracking	7978	1994	2.9499
	Ransomware	7751	1938	13.5420

IEC 60870-5-104 Intrusion Detection Dataset was published by Radoglou-Grammatikis et al. [30] for the evaluation of AI based intrusion detection systems (IDS), it was developed within two H2020 projects, ELECTRON: rEsilient and seLf-healed EleCTRical pOwer Nanogrid (101021936) and SDN-microSENSE: SDN - microgrid reSilient Electrical eNergy SystEm (833955).A network topology of seven industrial entities, a Human-Machine Interface (HMI), and three cyber attackers was used to construct this dataset. The data contains twelve cyberattacks categorized into unauthorized access, traffic sniffing, MITM, and DoS attacks.

It also include a normal traffic class. IEC 60870-5-104 Intrusion Detection Dataset contains eleven features including complete network configuration, traffic logs, attack diversity, heterogeneity, and a well-structured feature set. The data files include zip archives associated to entities and devices related to each attack, each archive includes PCAP traffic data, flow statistics from CICFlowMeter, and results from the Python parser. Table 3 and 4 introduce the class distribution of samples from the CICFlowMeter, and the Python parser across both training and test.

Table 2 Summary of class distribution of the IEC 60870-5-104- CICFlowMeter data.

Attack family	Attacks class	Training samples	Testing samples	Class weights
MITM	MITM	914	391	1.0
traffic sniffing	traffic sniffing	914	391	1.0
unauthorized access	C_RD_NA_1	914	391	1.0
	C_CI_NA_1	914	391	1.0
	C_RP_NA_1	914	391	1.0
	C_SC_NA_1	914	391	1.0
	C_SE_NA_1	914	391	1.0
DOS	M_SP_NA_1_DOS	914	391	1.0
	C_CI_NA_1_DOS	914	391	1.0
	C_SE_NA_1_DOS	914	391	1.0
	C_RD_NA_1_DOS	914	391	1.0
	C_RP_NA_1_DOS	914	391	1.0

Table 3 Summary of class distribution of the IEC 60870-5-104- Python parser data.

Attack family	Attacks class	Training samples	Testing samples	Class weights
MITM	MITM	914	391	1.0
traffic sniffing	traffic sniffing	914	391	1.0
unauthorized access	C_RD_NA_1	914	391	1.0
	C_CI_NA_1	914	391	1.0
	C_RP_NA_1	914	391	1.0
	C_SC_NA_1	914	391	1.0
	C_SE_NA_1	914	391	1.0
DOS	M_SP_NA_1_DOS	914	391	1.0
	C_CI_NA_1_DOS	914	391	1.0
	C_SE_NA_1_DOS	914	391	1.0
	C_RD_NA_1_DOS	914	391	1.0
	C_RP_NA_1_DOS	914	391	1.0

The DNP3 Intrusion Detection Dataset was tailored by Radoglou-Grammatikis et al. [31] as a benchmark centered on the Distributed Network Protocol Version 3 (DNP3) to enhance the performance of Intrusion Detection and Prevention Systems (IDPS).In this dataset, a network topology containing eight industrial entities including eight industrial entities, one Human Machine Interfaces (HMI) and three cyber attackers was used configure this dataset. The data encapsulate nine DNP3 related cyberattack scenarios related unauthorized command execution and DoS attacks, in addition to a normal attack class. The DNP3 Intrusion Detection Dataset contains eleven features including complete network configuration, detailed traffic logs, labeled datasets, comprehensive interaction data, full capture files, protocol diversity, attack variations, system heterogeneity, feature sets, and metadata. The dataset has nine attack folders each one contains PCAP files, CICFlowMeter files, and labeled DNP3 flow statistics generated using a custom Python parser. Table 5 and 6 present the class distribution of samples from the CICFlowMeter, and the Python parser across both training and test.

Table 4 Summary of class distribution of the DNP3 - CICFlowMeter data.

Attack family	Attacks class	Training samples	Testing samples	Class weights
DNP3 unauthorized commands	INIT_DATA	466	200	1.0
	DISABLE_UNSOLICITED	466	200	1.0
	WARM_RESTART	466	200	1.0
	REPLAY	466	200	1.0
	DNP3_ENUMERATE	466	200	1.0
	COLD_RESTART	466	200	1.0
	DNP3_INFO	466	200	1.0
	STOP_APP	466	200	1.0
DoS	MITM_DOS	466	200	1.0

Table 5 Summary of class distribution of the DNP3 - Python parser data.

Attack family	Attacks class	Training samples	Testing samples	Class weights
DNP3 unauthorized commands	INIT_DATA	466	200	1.0
	DISABLE_UNSOLICITED	466	200	1.0
	WARM_RESTART	466	200	1.0
	REPLAY	466	200	1.0
	DNP3_ENUMERATE	466	200	1.0
	COLD_RESTART	466	200	1.0
	DNP3_INFO	466	200	1.0
	STOP_APP	466	200	1.0
DoS	MITM_DOS	466	200	1.0

6.2| Evaluation Metrics

IoT Deep Learning Based IDS performance evaluated using a number of metrics including:

1. **Accuracy**

Calculates overall correctness, provide an explanation of results, but it can mislead on unbalanced data sets [25].

$$(TP + TN) / (TP + TN + FP + FN)$$

2. **Precision**

Calculates the correctness of a prediction of an attack, which shows how many of the traffic flaggings are correct, and not false alarms [23], [25].

$$(TP / (TP + FP))$$

3. **Recall**

Measures the ability of a system to detect all of the actual attacks that occurred [23].

$$(TP / (TP + FN))$$

4. **F1-Score**

Used to find a measure that balances precision and recall, allowing for a single metric that may be used to compare two models when both metrics are equally important [25].

$$(2 \times (Precision \times Recall) / (Precision + Recall))$$

5. **ROC Analysis and AUC**

Receiver Operating Characteristics (ROC) charts are used to display a classifiers performance at various thresholds by displaying the true positive rate versus the false positive rate. The area under the ROC curve (AUC) is a measure of the classifiers performance independent of the threshold selected. Values close to 1.0 indicate good separation between classes [24], [27].

Table 6 Summary of evaluation metrics for intrusion detection systems.

Metric	Formula	Interpretation
Accuracy	(TP + TN) / (TP + TN + FP + FN)	Overall correctness
Precision	TP / (TP + FP)	Reliability of positive predictions
Recall	TP / (TP + FN)	Coverage of actual positives
F1-Score	2 × (Precision × Recall) / (Precision + Recall)	Harmonic mean of precision and recall

6.3| Comparative Performance Analysis

This subsection presents comparative performance analysis of ten baseline architectures evaluated on three IoT intrusion detection datasets: Edge-IIoTset, IEC 60870-5-104, and DNP3.

Performance on Edge-IIoTset Dataset

Table 8 presents quantitative comparison of ten baseline models on the Edge-IIoTset dataset. DCNNBiLSTM records 97.56% accuracy, 97.56% F1-score, and 0.9996 AUC; HCRNNIDS records 97.34% F1-score; CNN-WDLSTM records 97.45% F1-score. Hybrid CNN-RNN architectures record 1-2 percentage points higher than single-architecture approaches in this evaluation.

Among single-architecture models, BiLSTM records 97.12% F1-score, DLSTM records 96.78% F1-score, and vanilla RNN records 95.27% F1-score. The 2.29 percentage point gap between DCNNBiLSTM and RNN illustrates performance variability across architected. RNN lower metrics caused by vanishing gradients, which limiting effective learning of long-term dependencies.

Table 7 Quantitative Comparison of the results of different baselines on the Edge-IIoTset dataset

Models	Accuracy	Precision	Recall	F1-score	AUC
CNN[20]	0.9690	0.9692	0.9690	0.9690	0.9995
BiLSTM[21]	0.9712	0.9713	0.9712	0.9712	0.9995
RNN[22]	0.9523	0.9541	0.9523	0.9527	0.9986
DLSTM[23]	0.9678	0.9679	0.9678	0.9678	0.9992
HCRNNIDS[24]	0.9734	0.9735	0.9734	0.9734	0.9996
DCNNBiLSTM[25]	0.9756	0.9756	0.9756	0.9756	0.9996
CNN–WDLSTM[26]	0.9745	0.9745	0.9745	0.9745	0.9996
GRU[27]	0.9601	0.9606	0.9601	0.9602	0.9991
AE-LSTM[32]	0.9689	0.9690	0.9689	0.9689	0.9992
SAAE-DNN[28]	0.9721	0.9722	0.9721	0.9721	0.9995

Performance on IEC 60870-5-104, and DNP3 datasets

Tables 9–12 show comparisons of the results from different baselines on the IEC 60870-5-104 data, as well as on the DNP3- data, using both protocol-specific parser features and CICFlowMeter-based flow statistics.

DCNNBiLSTM has much lower performance compared to Edge-IIoTset (F1-score 97.56% and F1-score 56% from Table 9) shows how performance decline occurs when models experience domain-specific industrial control protocols that contain insufficient discriminative information in standard network flow statistics.

The CNN-WDLSTM and AE-LSTM perform 55% accuracy and 56% F1-scores, while BiLSTM, DLSTM, and GRU have very poor performances at approximately 40-41% accuracy. This indicates that these recurrent architectures alone are unable to capture industrial protocol patterns without complementary spatial feature extraction via convolutional layers. The difference in performance between hybrid (55-56%) and single architecture recurrent models (40-41%) is 15-16 percentage points, much larger than the 2-3 percentage point differences in Edge-IIoTset. This suggests that the architecture used becomes much more important when working with challenging domain-specific datasets.

Table 8 Quantitative Comparison of the results of different baselines on the IEC 60870-5-104-CICFlowMeter dataset.

Models	Accuracy	Precision	Recall	F1-score	AUC
CNN[20]	0.53	0.61	0.53	0.54	0.90
BiLSTM[21]	0.41	0.44	0.41	0.38	0.83
RNN[22]	0.51	0.67	0.51	0.52	0.89
DLSTM[23]	0.40	0.44	0.40	0.39	0.83
HCRNNIDS[24]	0.54	0.61	0.54	0.54	0.90
DCNNBiLSTM[25]	0.56	0.61	0.56	0.57	0.90
CNN–WDLSTM[26]	0.55	0.61	0.55	0.56	0.90
GRU[27]	0.41	0.44	0.41	0.39s	0.83
AE-LSTM[32]	0.55	0.60	0.55	0.56	0.90
SAAE-DNN[28]	0.52	0.62	0.52	0.51	0.89

Protocol-specific feature extraction improves performance (Table 10). CNN-WDLSTM achieving 55% accuracy and 54% F1-score using custom IEC 60870-5-104 parser features that capture protocol-layer characteristics. Performance remains modest compared to Edge-IIoTset, reflecting complexity of industrial control protocol traffic characterized by limited traffic volume which requires deep domain expertise for effective feature engineering.

Table 9 Quantitative Comparison of the results of different baselines on the IEC 60870-5-104-Python parser dataset.

Models	Accuracy	Precision	Recall	F1-score	AUC
CNN[20]	0.54	0.65	0.54	0.53	0.90
BiLSTM[21]	0.53	0.64	0.53	0.51	0.90
RNN[22]	0.53	0.66	0.53	0.51	0.90
DLSTM[23]	0.53	0.62	0.53	0.51	0.90
HCRNNIDS[24]	0.54	0.62	0.54	0.54	0.90
DCNNBiLSTM[25]	0.54	0.63	0.54	0.53	0.91
CNN–WDLSTM[26]	0.55	0.61	0.55	0.54	0.91
GRU[27]	0.54	0.61	0.54	0.53	0.90
AE-LSTM[32]	0.54	0.62	0.54	0.53	0.92
SAAE-DNN[28]	0.53	0.64	0.53	0.51	0.93

DNP3 dataset evaluation shows higher metrics with CICFlowMeter features compared to IEC 60870-5-104, DNP3 attack patterns may exhibit distinctive flow-level signatures (Table 11). HCRNNIDS and DCNNBiLSTM record 91% accuracy, 91% recall, and 91% F1-score with 0.99-1.00 AUC under these experimental conditions. High metrics on DNP3 may relate to the protocol distinctive characteristics.

Table 10 Quantitative Comparison of the results of different baselines on the DNP3 - CICFlowMeter dataset.

Models	Accuracy	Precision	Recall	F1-score	AUC
CNN[20]	0.87	0.92	0.87	0.87	0.99
BiLSTM[21]	0.81	0.88	0.81	0.79	0.98
RNN[22]	0.87	0.92	0.87	0.86	0.99
DLSTM[23]	0.81	0.88	0.81	0.79	0.98
HCRNNIDS[24]	0.91	0.93	0.91	0.91	0.99
DCNNBiLSTM[25]	0.91	0.92	0.91	0.91	1.00
CNN–WDLSTM[26]	0.88	0.93	0.88	0.88	0.99
GRU[27]	0.82	0.89	0.82	0.80	0.98
AE-LSTM[32]	0.86	0.91	0.86	0.86	0.99
SAAE-DNN[28]	0.86	0.91	0.86	0.86	0.99

DNP3 parser features performance (Table 12), with CNN, RNN, HCRNNIDS, DCNNBiLSTM, and CNN-WDLSTM all achieving 94% accuracy, 95% precision, and 94% F1-score with perfect 1.00 AUC. The convergence of multiple architectures to a 94% performance indicates that protocol-aware feature engineering is able to extract discriminative attack signatures, as well as reduce the relative importance of architectural complexity when sufficient high quality domain specific features exist. BiLSTM shows poor performance (accuracy = 69%, f1 score = 67%), compared to other datasets in which it has performed well, demonstrating that the choice of architecture is interdependent with the representation of features and characteristics of the dataset.

Table 11 Quantitative Comparison of the results of different baselines on the DNP3 - Python parser dataset.

Models	Accuracy	Precision	Recall	F1-score	AUC
CNN[20]	0.94	0.95	0.94	0.94	1.00
BiLSTM[21]	0.69	0.75	0.69	0.67	0.96
RNN[22]	0.94	0.95	0.94	0.94	1.00
DLSTM[23]	0.72	0.82	0.72	0.71	0.93
HCRNNIDS[24]	0.94	0.95	0.94	0.94	1.00
DCNNBiLSTM[25]	0.94	0.95	0.94	0.94	1.00
CNN–WDLSTM[26]	0.94	0.95	0.94	0.94	1.00
GRU[27]	0.74	0.86	0.74	0.74	0.93
AE-LSTM[32]	0.93	0.95	0.93	0.93	1.00
SAAE-DNN[28]	0.91	0.94	0.91	0.91	0.95

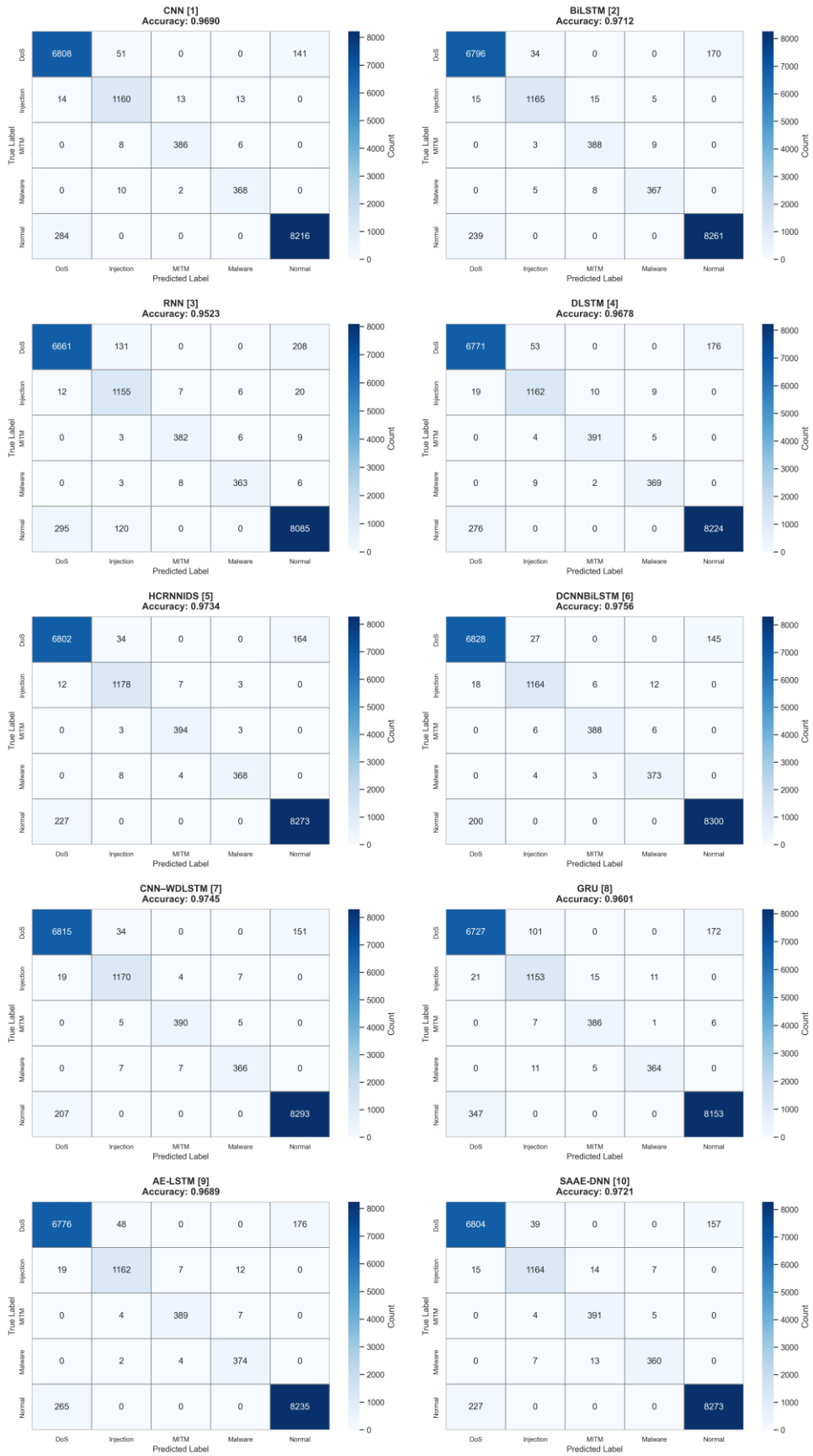
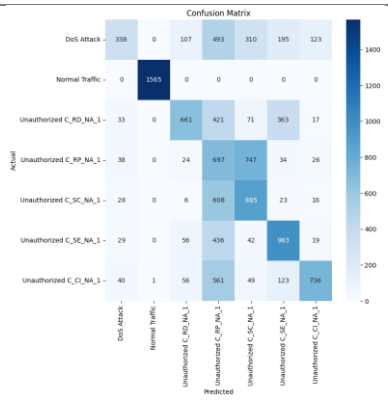
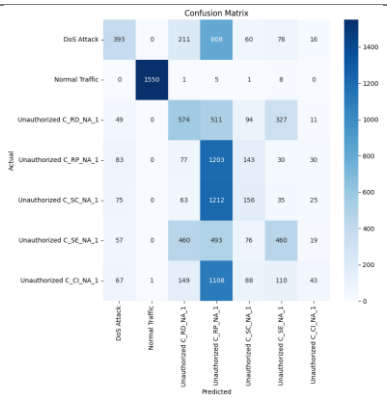


Figure 4 Confusion Matrices of different baselines on the Edge-IIoTset data.

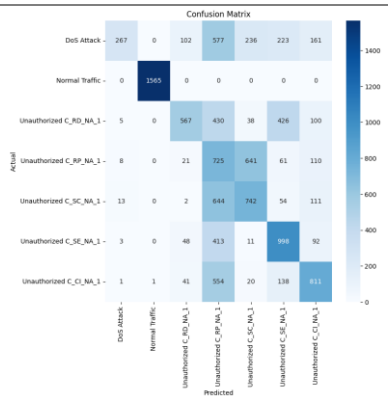
CNN[20]



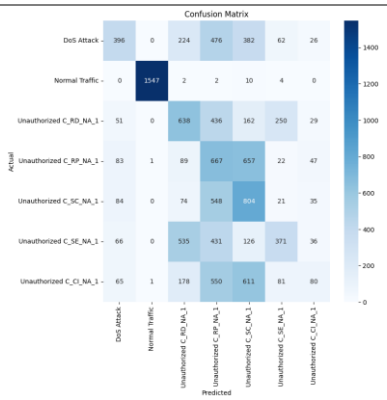
BiLSTM[21]



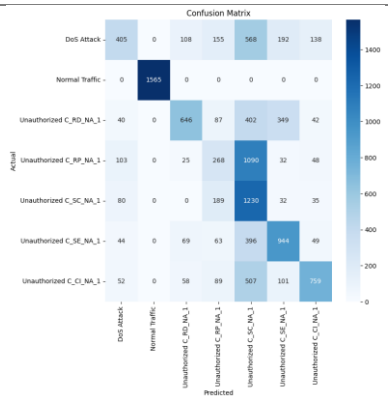
RNN[22]



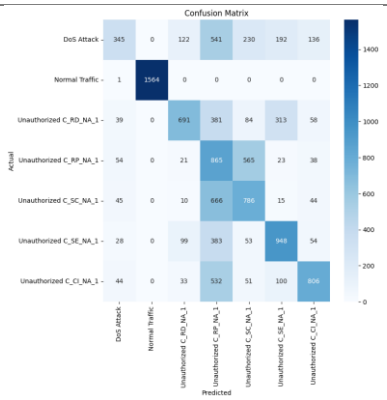
DLSTM[23]



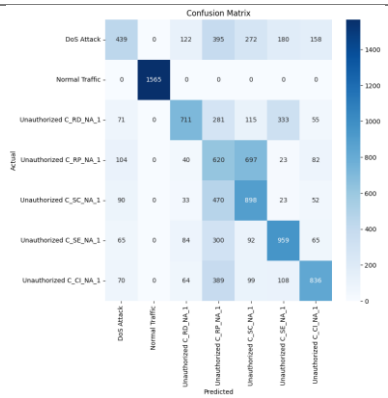
HCRNNIDS[24]



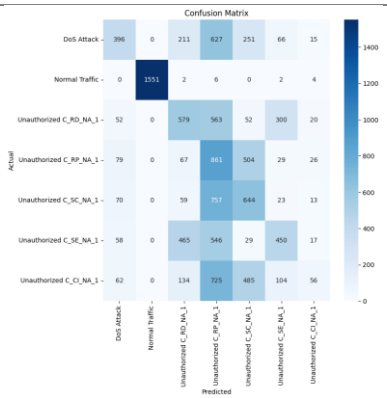
DCNNBiLSTM[25]



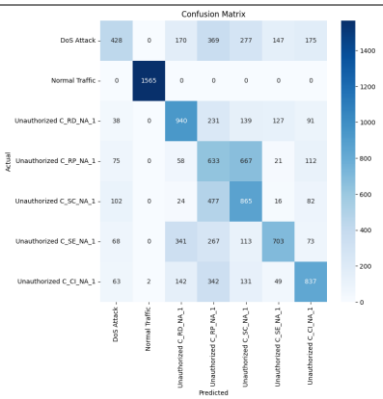
CNN–WDLSTM[26]



GRU[27]



AE-LSTM[32]



SAAE-DNN[28]

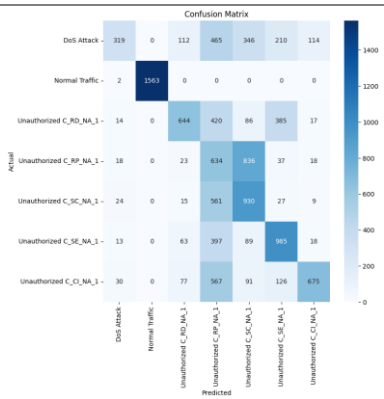
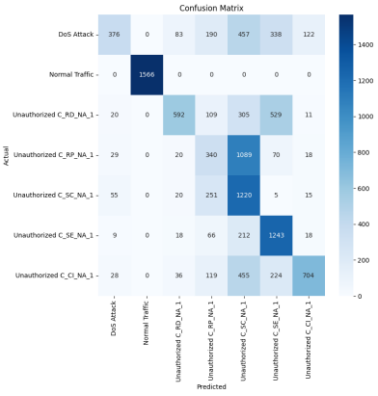
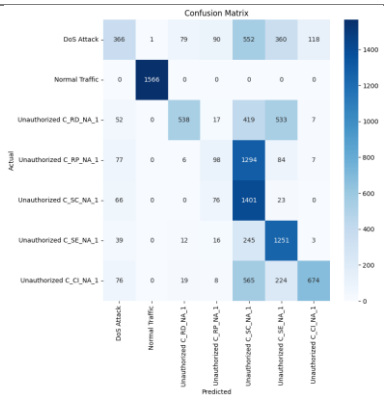


Figure 5 Confusion Matrices of different baselines on the IEC 60870-5-104- CICFlowMeter data.

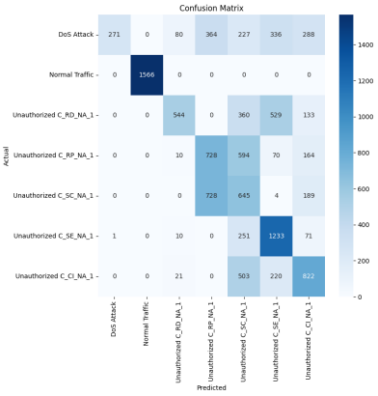
CNN[20]



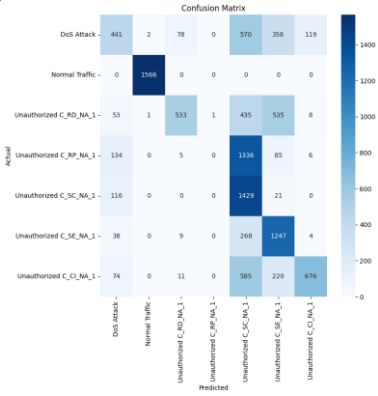
BiLSTM[21]



RNN[22]

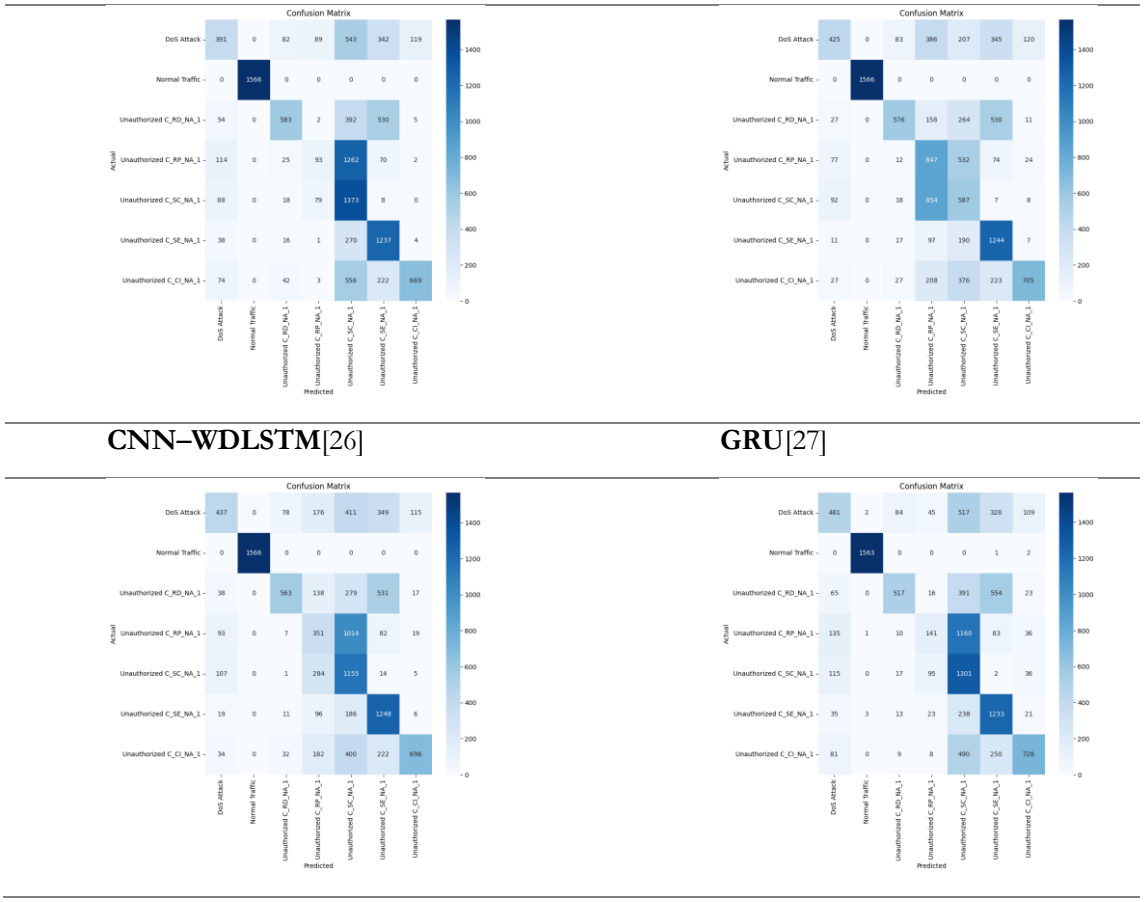


DLSTM[23]



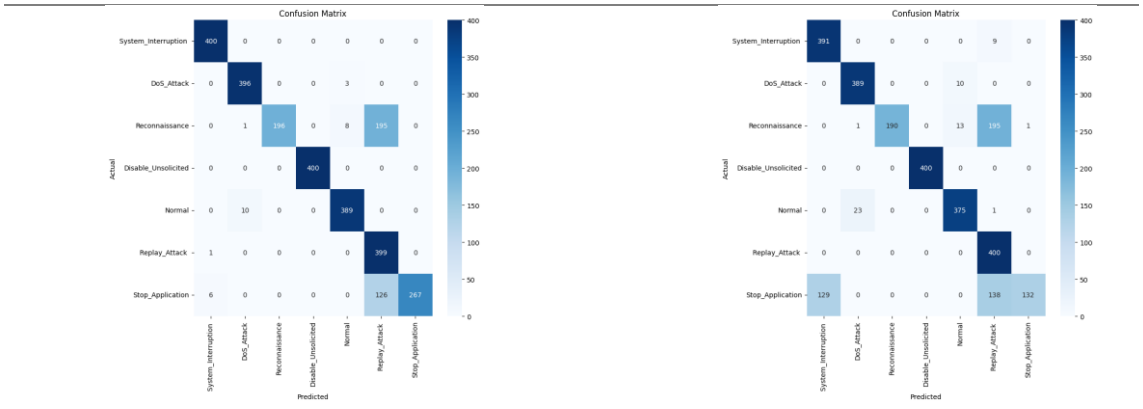
HCRNNIDS[24]

DCNNBiLSTM[25]



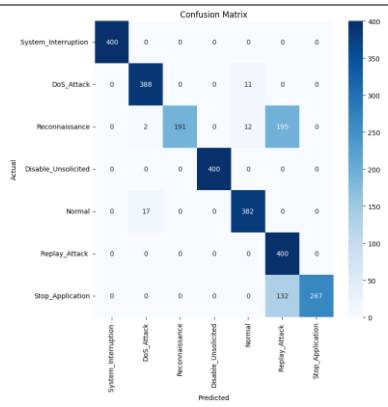
CNN–WDLSTM[26] **GRU**[27]

Figure 6 Confusion Matrices of different baselines on the IEC 60870-5-104- Python parser data.

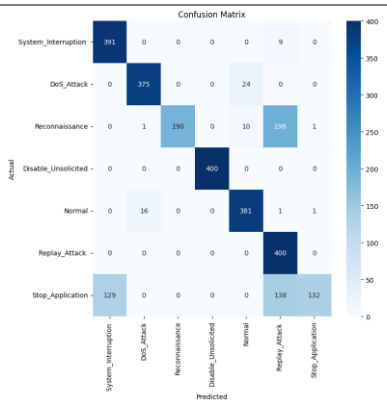


CNN[20] **BiLSTM**[21]

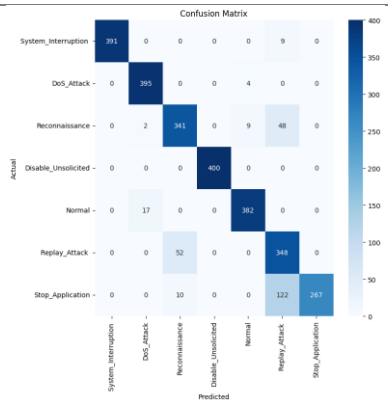
RNN[22] **DLSTM**[23]



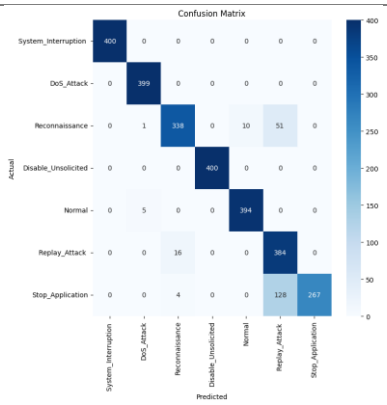
HCRNNIDS[24]



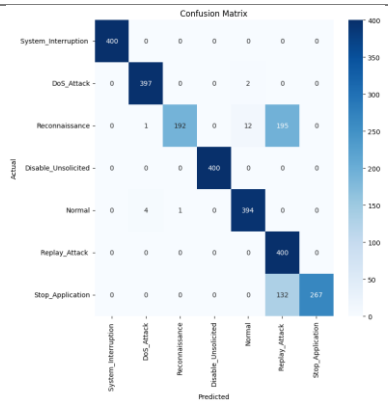
DCNNBiLSTM[25]



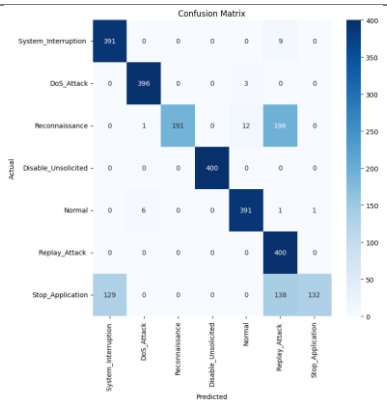
CNN–WDLSTM[26]



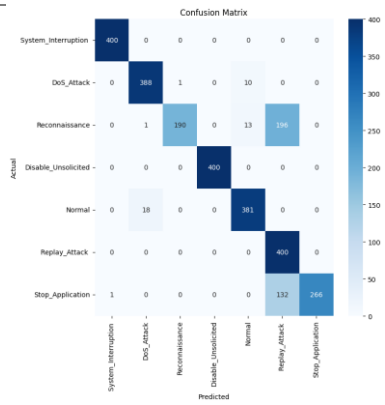
GRU[27]



AE-LSTM[32]



SAAE-DNN[28]



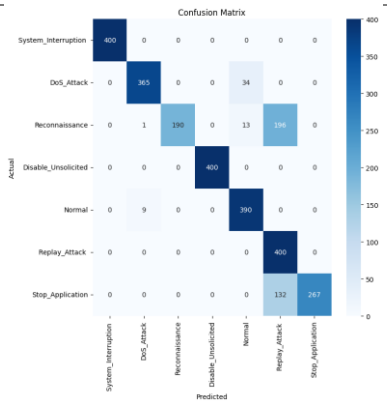
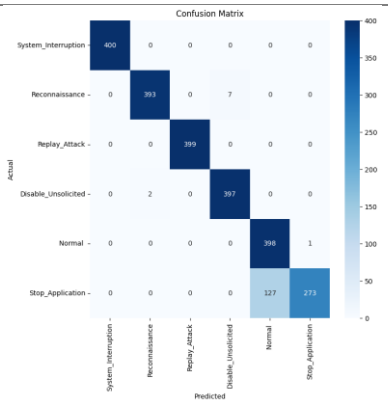
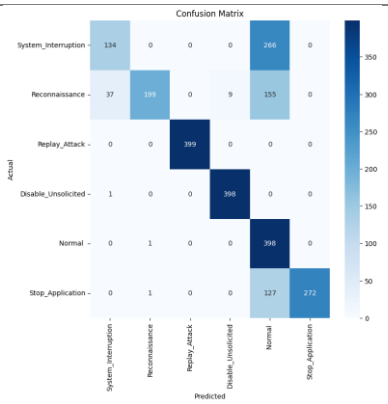


Figure 7 Confusion Matrices of different baselines on the DNP3 - CICFlowMeter data.

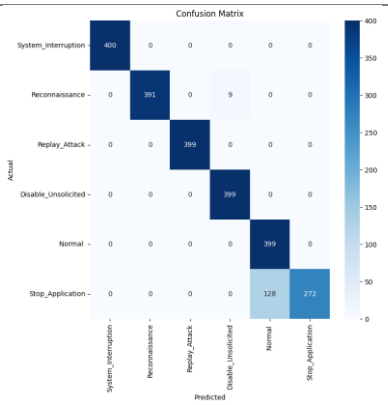
CNN[20]



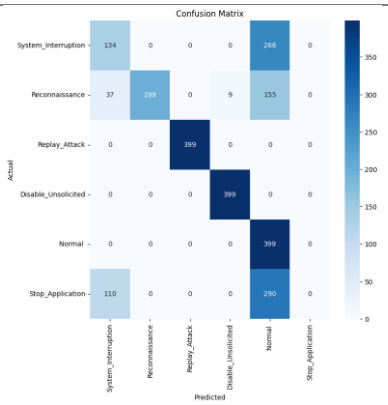
BiLSTM[21]



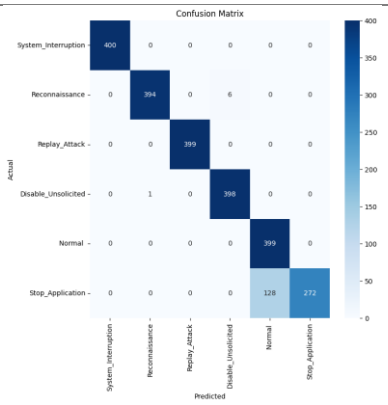
RNN[22]



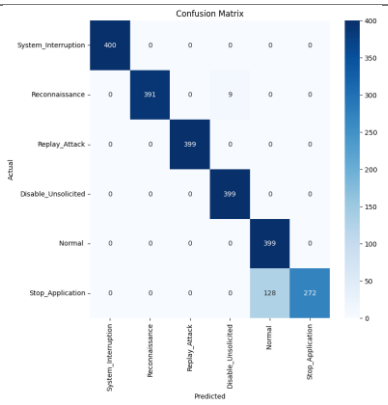
DLSTM[23]



HCRNNIDS[24]

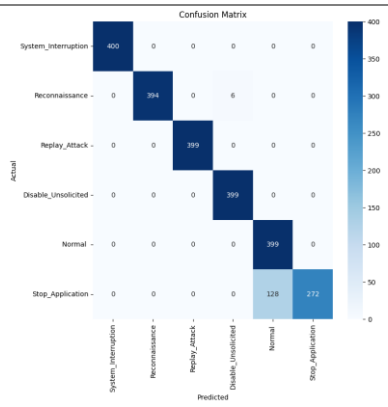


DCNNBiLSTM[25]

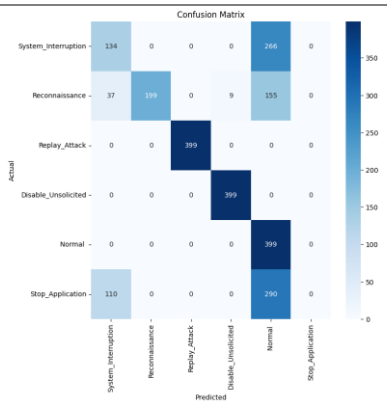


CNN–WDLSTM[26]

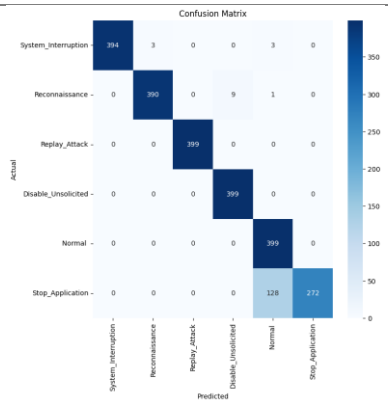
GRU[27]



AE-LSTM[32]



SAAE-DNN[28]



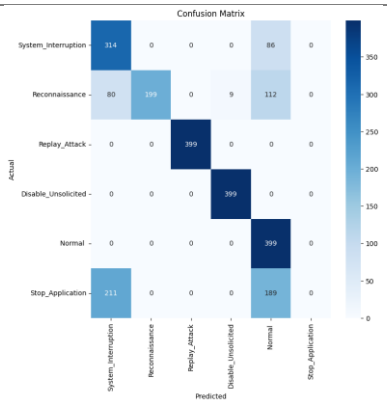


Figure 8 Confusion Matrices of different baselines on the DNP3 - Python parser data.

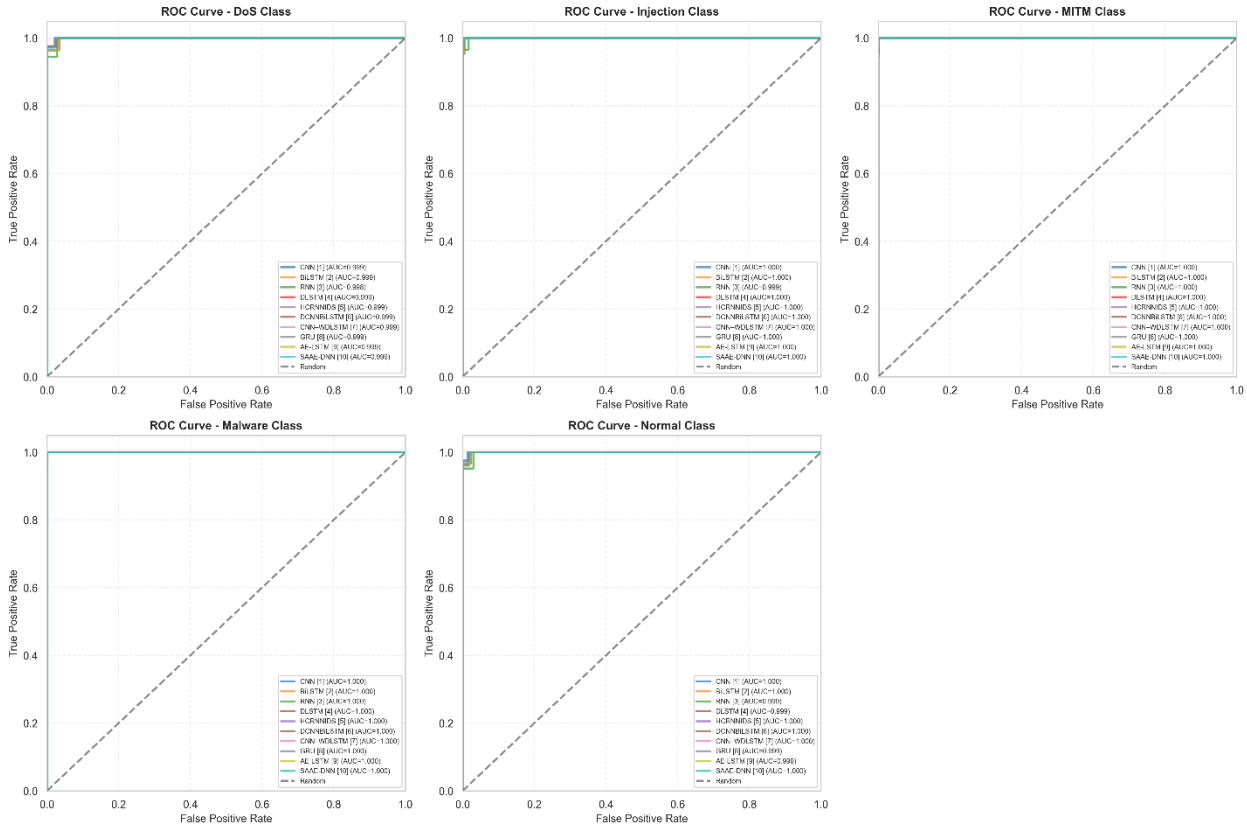


Figure 9 ROC curves of different baselines on the Edge-IIoTset data by class

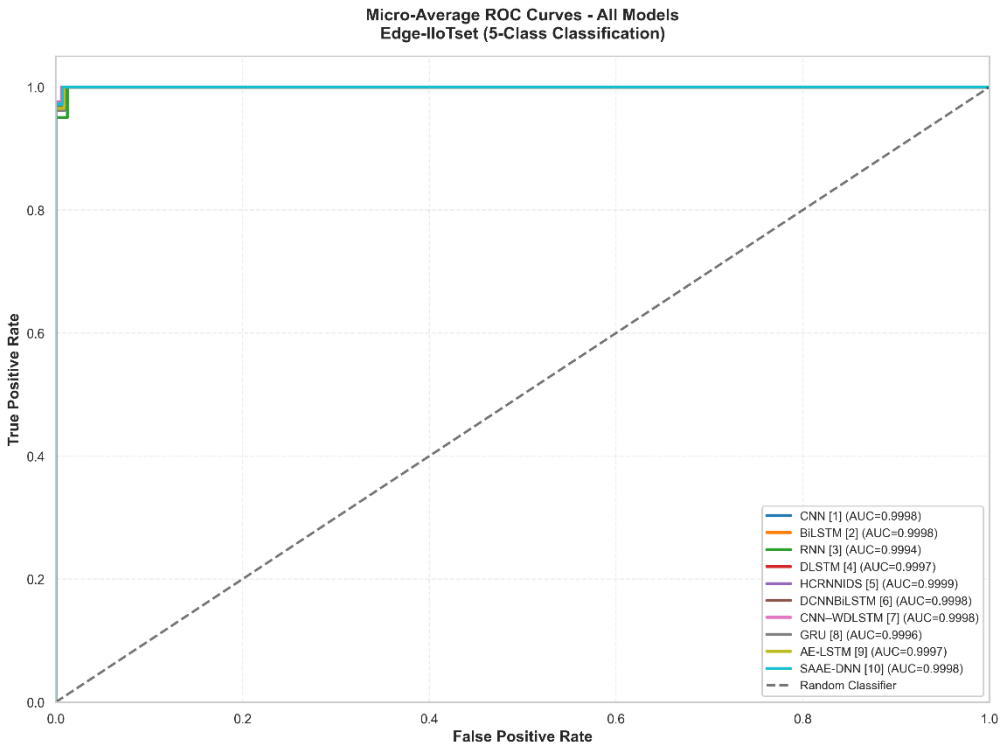
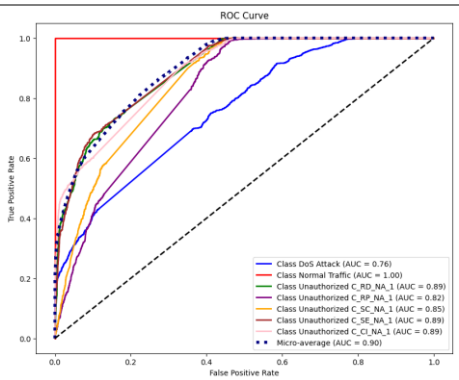
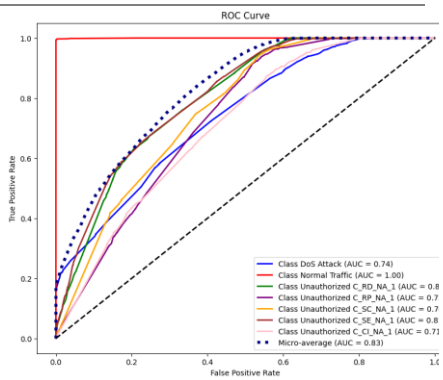


Figure 10 Micro-Average ROC curves of different baselines on the Edge-IIoTset data.

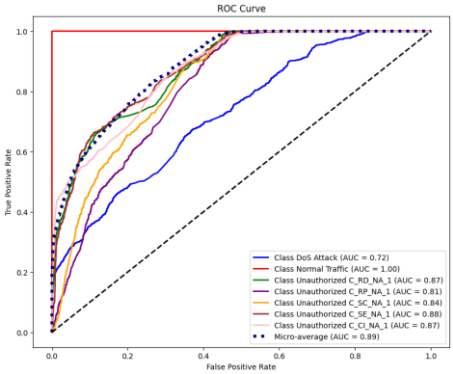
CNN[20]	**BiLSTM**[21]



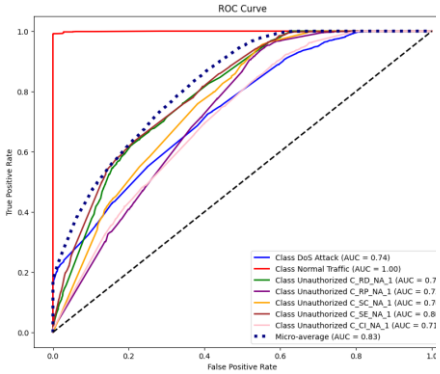
RNN[22]



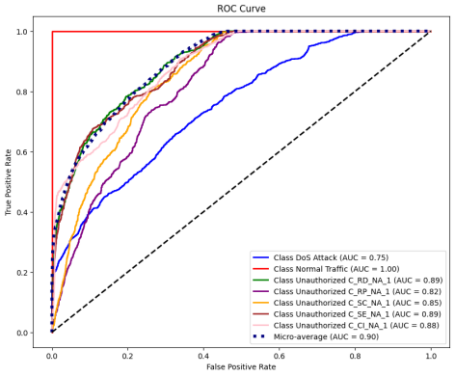
DLSTM[23]



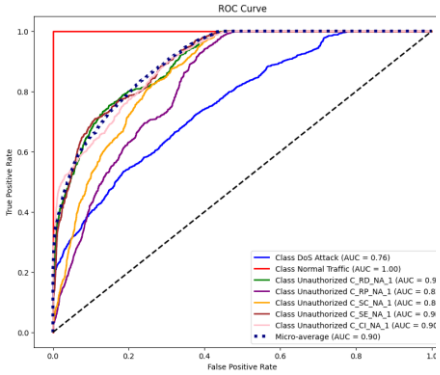
HCRNNIDS[24]



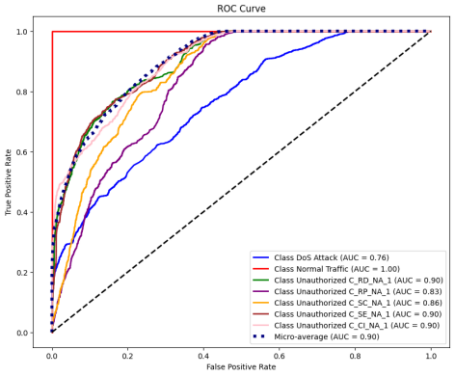
DCNNBiLSTM[25]



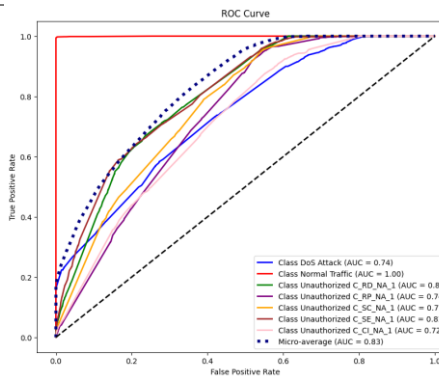
CNN–WDLSTM[26]



GRU[27]



AE-LSTM[32]



SAAE-DNN[28]

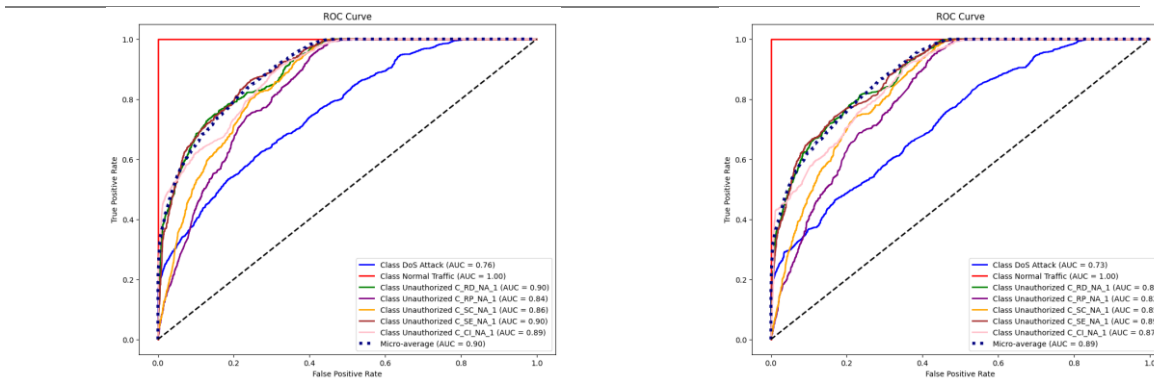
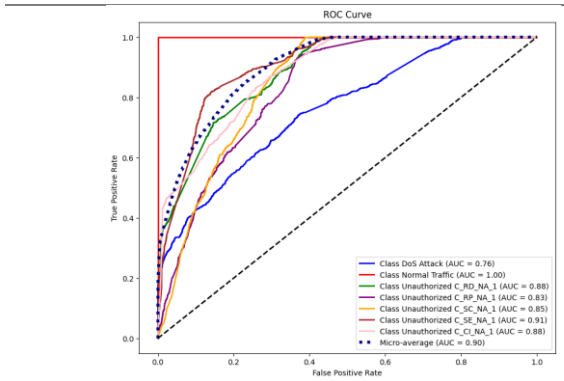
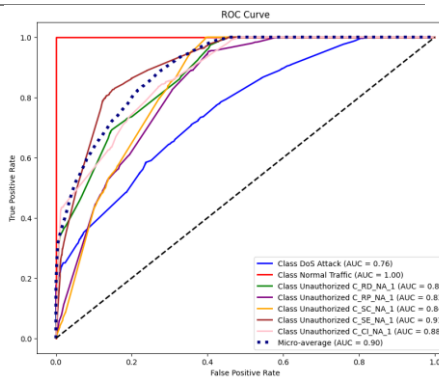


Figure 11 ROC curves of different baselines on the IEC 60870-5-104- CICFlowMeter data.

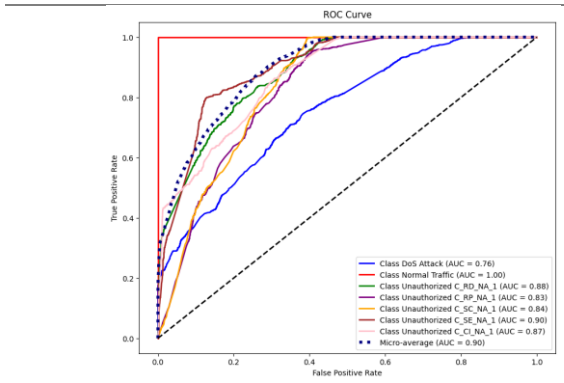
CNN[20]



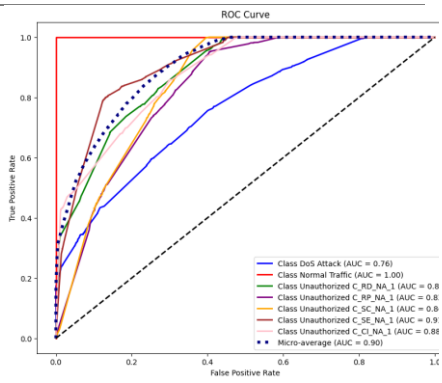
BiLSTM[21]



RNN[22]

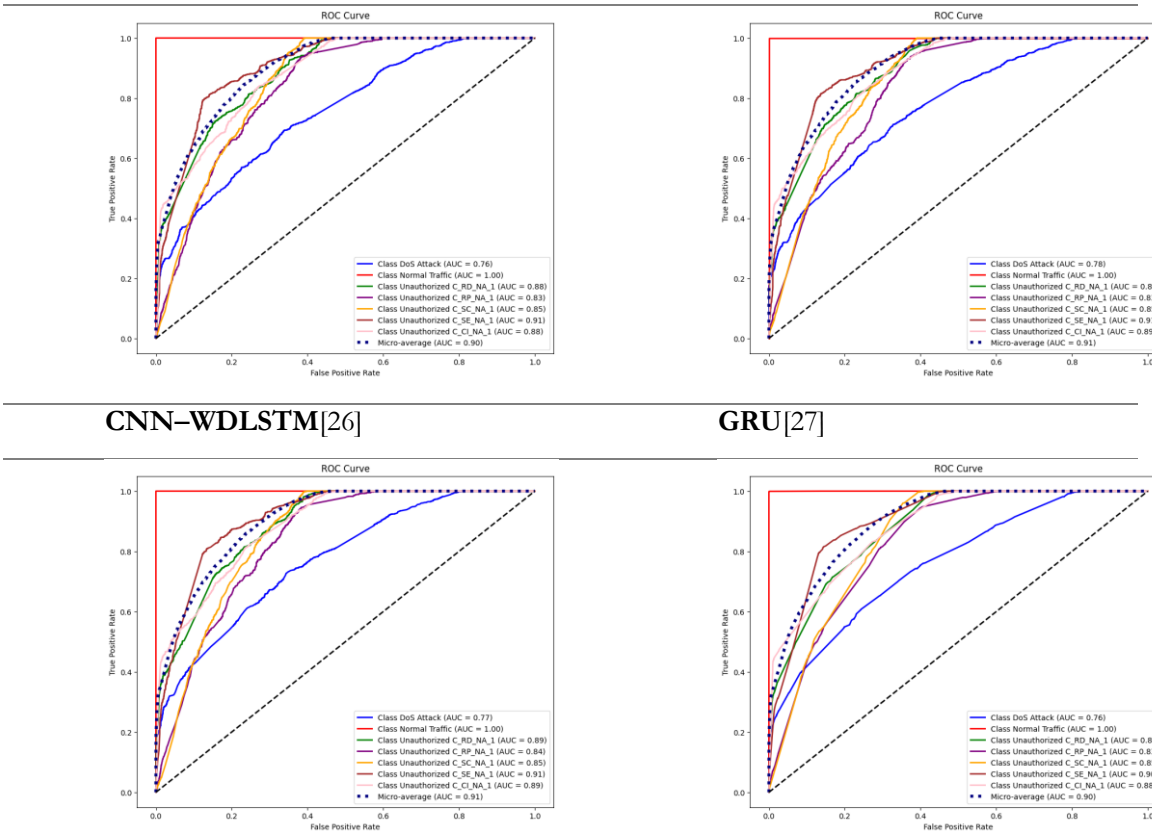


DLSTM[23]



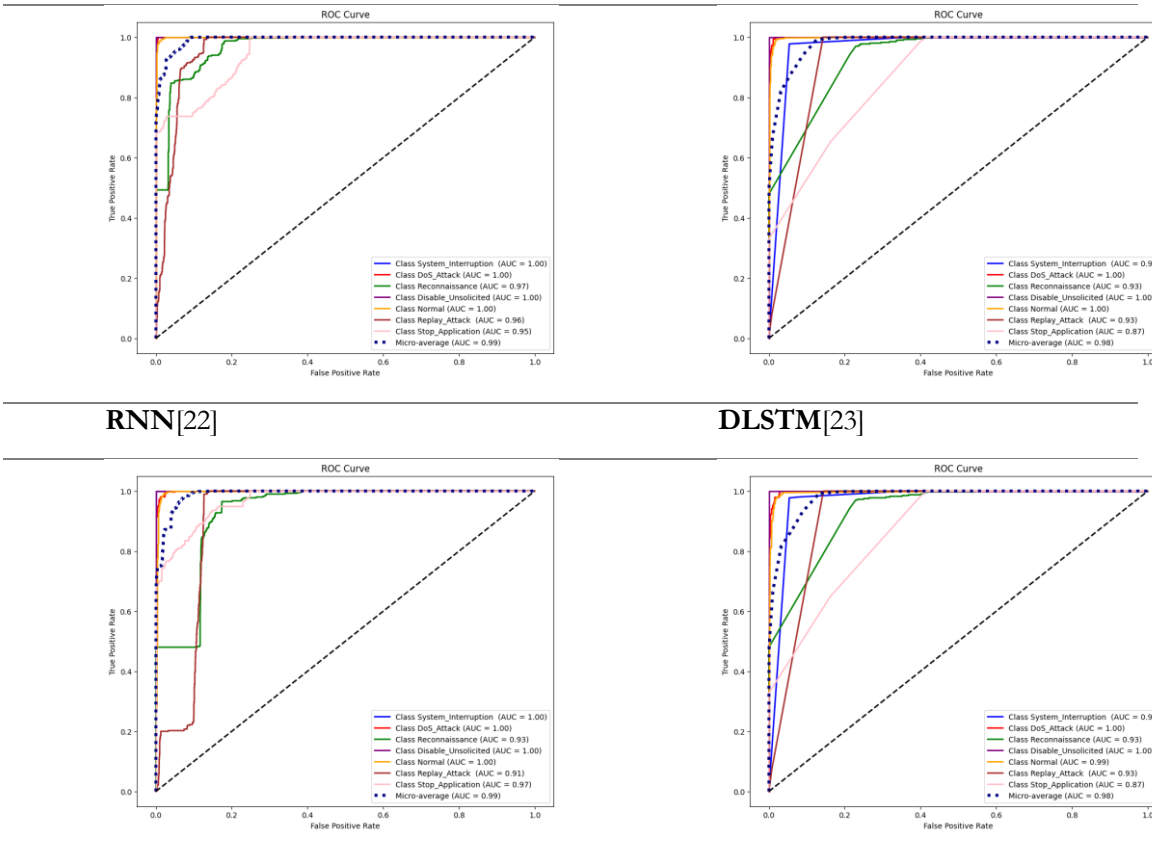
HCRNNIDS[24]

DCNNBiLSTM[25]



CNN–WDLSTM[26] **GRU**[27]

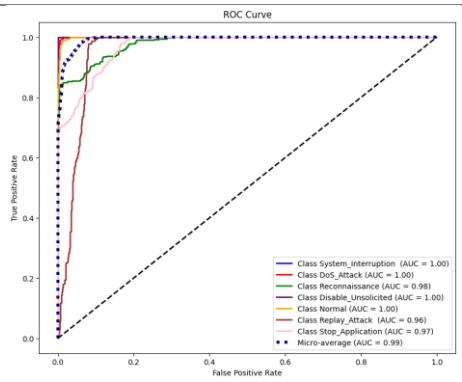
Figure 12 ROC curves of different baselines on the IEC 60870-5-104- Python parser data



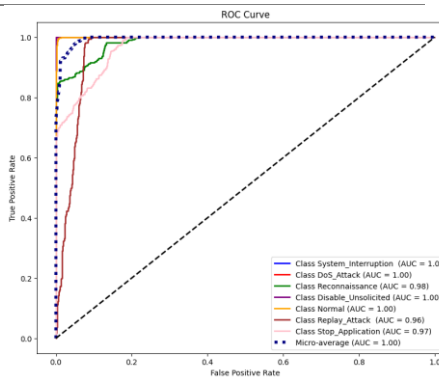
CNN[20] **BiLSTM**[21]

RNN[22] **DLSTM**[23]

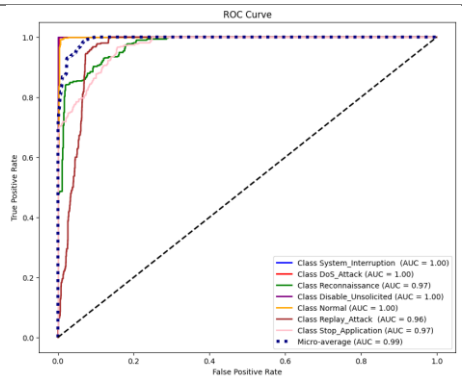
HCRNNIDS[24]



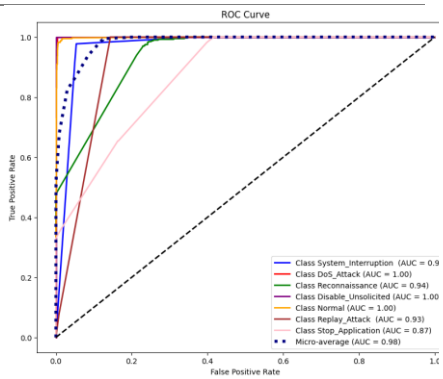
DCNNBiLSTM[25]



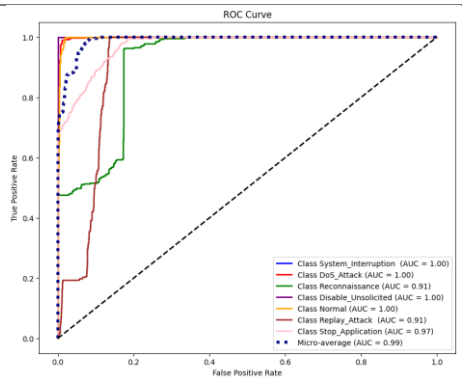
CNN–WDLSTM[26]



GRU[27]



AE-LSTM[32]



SAAE-DNN[28]

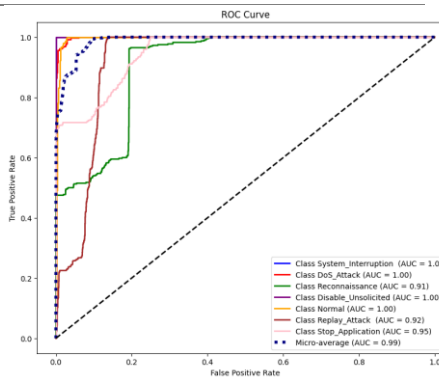
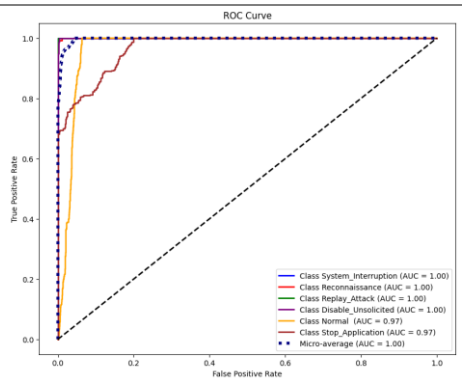
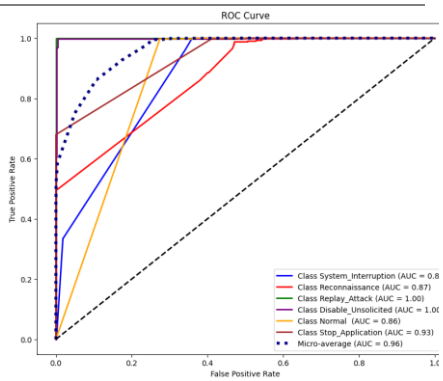


Figure 13 ROC curves of different baselines on the DNP3 - CICFlowMeter data.

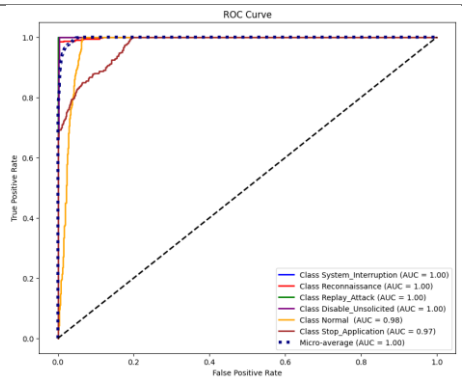
CNN[20]



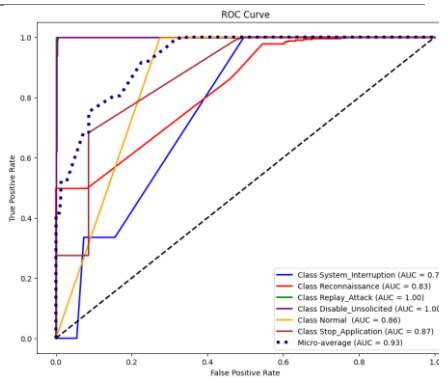
BiLSTM[21]



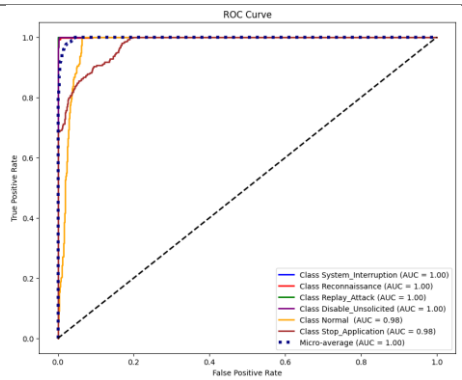
RNN[22]



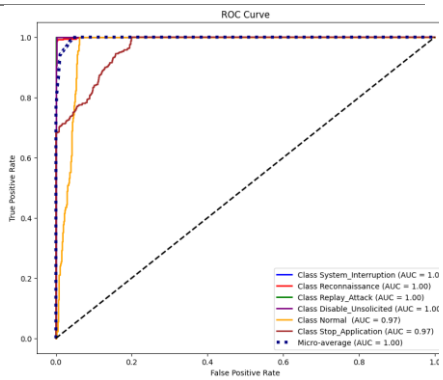
DLSTM[23]



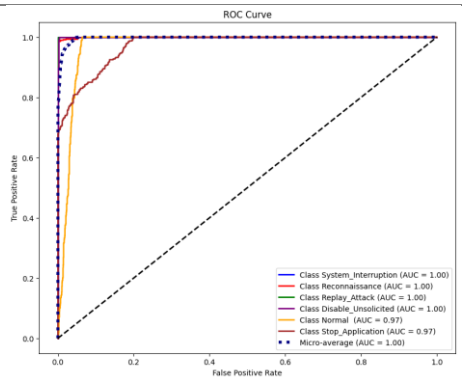
HCRNNIDS[24]



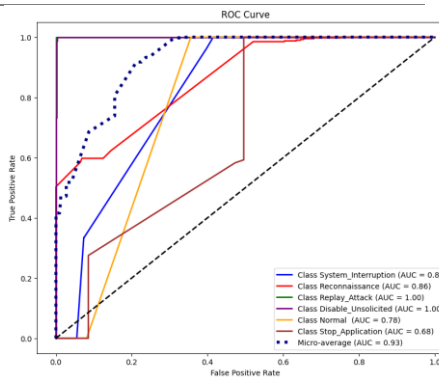
DCNNBiLSTM[25]



CNN–WDLSTM[26]



GRU[27]



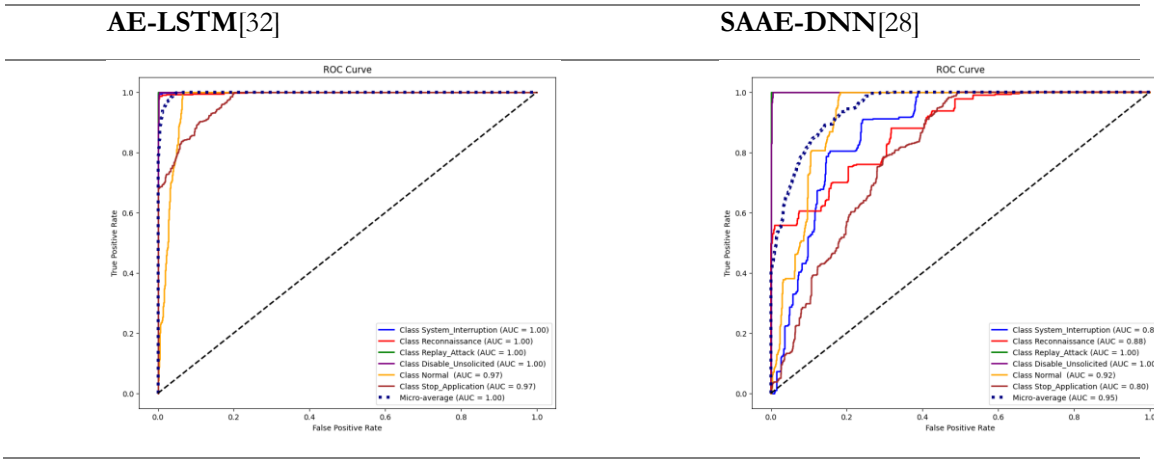


Figure 14. ROC curves of different baselines on the DNP3 - Python parser data.

7 | Discussion

In this section we analyses the findings of the systematic literature review to evaluate ongoing issues, as well as areas for future research

Intrusion Detection Systems have many issues relating to availably of dataset, the quality of those datasets and relevance to IDS different domains. Although there is a large amount of performance differences across all of the testing scenarios. Comparative analysis shows metric differences when models are evaluated across domains, DCNNBiLSTM records 97.56% F1-score on Edge-IIoTset but only 56-57% on IEC 60870-5-104 with CICFlowMeter features, representing a 40-percentage-point difference illustrating generalization challenges. Similarly, BiLSTM archives high Edge-IIoTset metrics 97.12% F1-score and records lower results on DNP3 protocol-specific features (69% accuracy, 67% F1-score), suggesting architectural choices interact with feature representations and protocol characteristics in complex ways.

IoT-specific datasets remain scarce, with Edge-IIoTset, IEC 60870-5-104, and DNP3 exceptions addressing consumer IoT, power grid communications, and SCADA systems. Even specialized datasets exhibit limitations such as class imbalance, limited temporal coverage, and controlled settings that misses operational noise and protocol implementation variations.

IoT devices operate under severe computational, memory, and energy constraints that challenge deployment of sophisticated deep learning models. Models deployment needs resource requirements for high-complexity architectures, hybrid models requiring edge AI accelerators or cloud deployment.

Effective intrusion detection in IoT environments demands real-time processing capabilities enabling timely threat response. Critical infrastructure applications require low latency to enable protective relay actions before failures. Hybrid architectures recording higher accuracy metrics which impose higher latencies than lightweight models. Only lightweight architectures may satisfy latency constraints, and accepting some accuracy penalty.

Systematic comparative analysis across diverse datasets presents consistent architectural patterns influencing detection metrics, providing evidence for architecture selection in operational deployments. Hybrid CNN-LSTM architectures rank among top scorers regardless of dataset characteristics, DCNNBiLSTM records rank 1 on Edge-IIoTset (97.56% F1), rank 1 on IEC 60870-5-104 CICFlowMeter (57% F1), rank 1 on DNP3 CICFlowMeter (91% F1), and rank 1 on DNP3 parser features (94% F1), indicating consistent cross-dataset performance. HCRNNIDS and CNN-WDLSTM maintain top three rankings across four of five evaluation scenarios, indicating that hybrid architectures combining CNN spatial feature extraction with RNN temporal modeling address diverse attack.

Metric differences between hybrid architectures vary across datasets, providing insights about when architectural representation appears to be effective. On Edge-IIoTset, the gap between DCNNBiLSTM (97.56% F1) and BiLSTM (97.12% F1) reaches only 0.44 percentage points, show that architectural complexity is regarded when datasets contain strong discriminative features. On IEC 60870-5-104 CICFlowMeter features, hybrid architectures (DCNNBiLSTM: 56%, CNN-WDLSTM: 55%) record 14-16 percentage points higher than recurrent-only models (BiLSTM: 41%, DLSTM: 40%, GRU: 41%), indicating architectural sophistication appears more beneficial for challenging domains where generic features provide limited discriminative power.

Recurrent only architectures shows highly variable metrics. BiLSTM records 97.12% F1-score on Edge-IIoTset, 53% accuracy on IEC 60870-5-104 parser features, 69% accuracy on DNP3 parser features, prove that temporal models require careful feature for consistent results. Vanilla RNN records lower metrics across all scenarios (Edge-IIoTset: 95.27%, IEC CICFlowMeter: 51%, DNP3 CICFlowMeter: 87%, DNP3 parser: 94%), that causes by vanishing gradient limitations, protocol-specific DNP3 features relate to shorter temporal dependencies in that dataset. Autoencoder-based approaches (AE-LSTM, SAAE-DNN) demonstrate consistent metrics across all datasets, prove that unsupervised pretraining may provide stable initialization reducing overfitting risk.

8 | Conclusion and Future Work

Our comparative study evaluate both the performance of ten different deep learning architectures and the relative merits of each model using a combination of three IoT-specific datasets Edge-IIoTset, IEC 60870-5-104, and DNP3. The results of this comparison provide insights into architectures in terms of the degree to which they generalize to other application contexts.

CNN-LSTM hybrid architectures performed better than non-hybrid architectures in all cases, CNN-BiLSTM outperformed the baseline CNN architectures at all three datasets, Edge-IIoTset with a 97.56% F1-score, IEC 60870-5-104 with a 56-57% F1-score, and DNP3 with a 91-94% F1-score, the CNN-BiLSTM architecture also outperformed the baseline architectures at all three datasets depending on the specific characteristics of the dataset used.The BiLSTM model has shown strong performance on the Edge- IIoTset dataset (97.12% accuracy) but poor performance on the DNP3 dataset (69% accuracy). These results illustrate the complexity of interactions between the features of data and the architecture of deep learning models. Models that were able to detect anomalies in IoT traffic at high F1-scores than others are likely to exceed the processing capabilities of most IoT devices. Therefore, many of the models need to be deployed with edge AI accelerators. Energy constraints are still one of the primary limiting factors for the deployment of sophisticated models in battery powered IoT applications.

There are four major areas that require additional study, The development of architectures optimized for edge constraints using neural architecture search to improve F1-scores under resource limited conditions, Cross-domain transfer learning, Explainable AI methods that interpret the errors made by anomaly detection models, Development of comprehensive IoT datasets with emerging protocols and testing anomaly detection models in the presence of adversarial attacks.

Funding

This research has no funding source.

Conflicts of Interest

The authors declare that there is no conflict of interest in the research.

Ethical Approval

This article does not contain any studies with human participants or animals performed by any of the authors.

Reference

[1] baraa I. Farhan and Ammar D.Jasim, "A Survey of Intrusion Detection Using Deep Learning in Internet of Things," *Iraqi J. Comput. Sci. Math.*, pp. 83–93, Jan. 2022, doi: 10.52866/ijcsm.2022.01.01.009.

[2] Z. Ahmad, A. Shahid Khan, C. Wai Shiang, J. Abdullah, and F. Ahmad, "Network intrusion detection system: A systematic study of machine learning and deep learning approaches," *Trans. Emerg. Telecommun. Technol.*, vol. 32, no. 1, Jan. 2021, doi: 10.1002/ett.4150.

[3] K. Albulayhi, A. A. Smadi, F. T. Sheldon, and R. K. Abercrombie, "IoT Intrusion Detection Taxonomy, Reference Architecture, and Analyses," *Sensors*, vol. 21, no. 19, p. 6432, Sep. 2021, doi: 10.3390/s21196432.

[4] J. Mohanty, S. Mishra, S. Patra, B. Pati, and C. R. Panigrahi, "IoT Security, Challenges, and Solutions: A Review," 2021, pp. 493–504. doi: 10.1007/978-981-15-6353-9_46.

[5] P. L. S. Jayalaxmi, R. Saha, G. Kumar, M. Conti, and T.-H. Kim, "Machine and Deep Learning Solutions for Intrusion Detection and Prevention in IoTs: A Survey," *IEEE Access*, vol. 10, pp. 121173–121192, 2022, doi: 10.1109/ACCESS.2022.3220622.

[6] S. W. Lee *et al.*, "Towards secure intrusion detection systems using deep learning techniques: Comprehensive analysis and review," 2021. doi: 10.1016/j.jnca.2021.103111.

[7] J. Asharf, N. Moustafa, H. Khurshid, E. Debie, W. Haider, and A. Wahab, "A review of intrusion detection systems using machine and deep learning in internet of things: Challenges, solutions and future directions," 2020. doi: 10.3390/electronics9071177.

[8] M. Azrour, J. Mabrouki, A. Guezzaz, and A. Kanwal, "Internet of Things Security: Challenges and Key Issues," *Secur. Commun. Networks*, vol. 2021, pp. 1–11, Sep. 2021, doi: 10.1155/2021/5533843.

[9] M. Nuaimi, L. C. Fourati, and B. Ben Hamed, "Intelligent approaches toward intrusion detection systems for Industrial Internet of Things: A systematic comprehensive review," 2023. doi: 10.1016/j.jnca.2023.103637.

[10] A. Thakkar and R. Lohiya, "A survey on intrusion detection system: feature selection, model, performance measures, application perspective, challenges, and future research directions," *Artif. Intell. Rev.*, 2022, doi: 10.1007/s10462-021-10037-9.

[11] D. Chou and M. Jiang, "A Survey on Data-driven Network Intrusion Detection," *ACM Comput. Surv.*, vol. 54, no. 9, pp. 1–36, Dec. 2022, doi: 10.1145/3472753.

[12] N. Sun *et al.*, "Cyber Threat Intelligence Mining for Proactive Cybersecurity Defense: A Survey and New Perspectives," *IEEE Commun. Surv. Tutorials*, 2023, doi: 10.1109/COMST.2023.3273282.

[13] A. Awadallah *et al.*, "Artificial Intelligence-Based Cybersecurity for the Metaverse: Research Challenges and Opportunities," *IEEE Commun. Surv. Tutorials*, pp. 1–1, 2024, doi: 10.1109/COMST.2024.3442475.

[14] H. Kheddar, D. W. Dawoud, A. I. Awad, Y. Himeur, and M. K. Khan, "Reinforcement-Learning-Based Intrusion Detection in Communication Networks: A Review," *IEEE Commun. Surv. Tutorials*, pp. 1–1, 2024, doi: 10.1109/COMST.2024.3484491.

[15] A. Nascita, G. Aceto, D. Ciuonzo, A. Montieri, V. Persico, and A. Pescapé, "A Survey on Explainable Artificial Intelligence for Internet Traffic Classification and Prediction, and Intrusion Detection," *IEEE Commun. Surv. Tutorials*, pp. 1–1, 2024, doi: 10.1109/COMST.2024.3504955.

[16] B. Lampe and W. Meng, "Intrusion Detection in the Automotive Domain: A Comprehensive Review," *IEEE Commun. Surv. Tutorials*, 2023, doi: 10.1109/COMST.2023.3309864.

[17] M. Zipperle, F. Gottwalt, E. Chang, and T. Dillon, "Provenance-based Intrusion Detection Systems: A Survey," *ACM Comput. Surv.*, 2022, doi: 10.1145/3539605.

[18] J. Halvorsen, C. Izurieta, H. Cai, and A. Gebremedhin, "Applying Generative Machine Learning to Intrusion Detection: A Systematic Mapping Study and Review," *ACM Comput. Surv.*, vol. 56, no. 10, pp. 1–33, Oct. 2024, doi: 10.1145/3659575.

[19] C. Wohlin, "Guidelines for snowballing in systematic literature studies and a replication in software engineering," in *ACM International Conference Proceeding Series*, 2014. doi: 10.1145/2601248.2601268.

[20] W. H. Aljuaid and S. S. Alshamrani, "A Deep Learning Approach for Intrusion Detection Systems in Cloud Computing Environments," *Appl. Sci.*, vol. 14, no. 13, p. 5381, Jun. 2024, doi: 10.3390/app14135381.

[21] Y. Imrana, Y. Xiang, L. Ali, and Z. Abdul-Rauf, "A bidirectional LSTM deep learning approach for intrusion detection," *Expert Syst. Appl.*, 2021, doi: 10.1016/j.eswa.2021.115524.

[22] Y. C. Wang, Y. C. Houng, H. X. Chen, and S. M. Tseng, "Network Anomaly Intrusion Detection Based on Deep Learning Approach," *Sensors*, 2023, doi: 10.3390/s23042171.

[23] S. M. Kasongo and Y. Sun, "A Deep Long Short-Term Memory based classifier for Wireless Intrusion Detection System," *ICT Express*, 2020, doi: 10.1016/j.icte.2019.08.004.

[24] M. A. Khan, "HCRNNIDS: Hybrid convolutional recurrent neural network-based network intrusion detection system," *Processes*, 2021, doi: 10.3390/pr9050834.

[25] V. Hnamte and J. Hussain, "DCNNBiLSTM: An Efficient Hybrid Deep Learning-Based Intrusion Detection System," *Telemat. Informatics Reports*, 2023, doi: 10.1016/j.teler.2023.100053.

[26] M. M. Hassan, A. Gumaei, A. Alsanad, M. Alrubaian, and G. Fortino, "A hybrid deep learning model for efficient intrusion detection in big data environment," *Inf. Sci. (Ny).*, 2020, doi: 10.1016/j.ins.2019.10.069.

[27] M. S. Ansari, V. Bartoš, and B. Lee, "GRU-based deep learning approach for network intrusion alert prediction," *Futur. Gener. Comput. Syst.*, 2022, doi: 10.1016/j.future.2021.09.040.

[28] C. Tang, N. Luktarhan, and Y. Zhao, "Saae-dnn: Deep learning method on intrusion detection," *Symmetry (Basel).*, 2020, doi: 10.3390/sym12101695.

[29] M. A. Ferrag, O. Friha, D. Hamouda, L. Maglaras, and H. Janicke, "Edge-IIoTset: A New Comprehensive Realistic Cyber Security Dataset of IoT and IIoT Applications for Centralized and Federated Learning," *IEEE Access*, 2022, doi: 10.1109/ACCESS.2022.3165809.

[30] P. Radoglou-Grammatikis *et al.*, "Modeling, Detecting, and Mitigating Threats against Industrial Healthcare Systems: A Combined Software Defined Networking and Reinforcement Learning Approach," *IEEE Trans. Ind. Informatics*, 2022, doi: 10.1109/TII.2021.3093905.

[31] V. Kelli *et al.*, "Attacking and Defending DNP3 ICS/SCADA Systems," in *Proceedings - 18th Annual International Conference on Distributed Computing in Sensor Systems, DCOSS 2022*, 2022. doi: 10.1109/DCOSS54816.2022.00041.

[32] E. Mushtaq, A. Zameer, M. Umer, and A. A. Abbasi, "A two-stage intrusion detection system with auto-encoder and LSTMs," *Appl. Soft Comput.*, 2022, doi: 10.1016/j.asoc.2022.108768.