IJARIIT INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X
Impact Factor: 6.078
(Volume 11, Issue 5 - V11I5-1314)
Available online at: https://www.ijariit.com

A Comparative Study of Machine Learning Models for Predictive Analytics in Detecting Security Breaches Across Industrial IoT-Based Critical Infrastructure in U.S. Organizations

Tolulope Onasanya
tdonasanya@aggies.ncat.edu
North Carolina Agricultural and Technical State University, North Carolina

Adeogo Olajide
aola5395@students.vsu.edu
Virginia State University, Ettrick, Virginia

Oduwunmi Odukoya
odukoyao@etsu.edu
East Tennessee State University, Johnson City, Tennessee

Hannah I. Tanimowo
tanim006@d.umn.edu
University of Minnesota Duluth, Duluth, Minnesota

ABSTRACT

Industrial Internet of Things (IIoT) technologies have emerged with significant security challenges due to increasing interconnectivity and network complexity. Thus, this study proposes and tests a centralized deep-learning intrusion detection system (IDS) particularly designed for IIoT networks. Convolutional Neural Networks (CNNs) and Multilayer Perceptrons (MLPs) were implemented in PyTorch and TensorFlow, and their performance was assessed on the CIC IoT-DIAD 2024 dataset to simulate real industrial traffic and varied attack vectors. Accuracy, precision, recall, F1-score, and confusion matrices were used for the evaluation metrics. Results showed that the TensorFlow CNN achieved the highest detection rate (80.1%), followed very closely by the PyTorch CNN (77.3%), highlighting the superior performance of CNN architectures, especially on TensorFlow, in recognizing intricate spatial patterns in IIoT traffic. The comparative research enhances IIoT cybersecurity research by identifying efficient deep learning models for intrusion detection and proposing a framework for adoption in industrial systems to improve resilience and mitigate vulnerability to advanced cyberattacks.

Keywords: *IIoT, Critical Infrastructure, Cyberattack, Deep learning, Pytorch, TensorFlow.*

1. INTRODUCTION

The Industrial Internet of Things (IIoT) has transformed industry operations significantly by integrating intelligent sensors, real-time analysis, and automated equipment in manufacturing plants. The technologies improve efficiency, facilitate predictive maintenance, and optimize data-driven decision-making, leading to accelerated, optimized processes [34], [58]. The growth of networked devices has also increased vulnerabilities in infrastructures. As industrial systems become increasingly interconnected, they are more vulnerable to cyberattacks, underscoring the need for strong security controls in IIoT implementations [62].

IIoT networks are highly susceptible to cyberattacks, including denial-of-service, data injection, man-in-the-middle, and malware propagation. Attacks exploit vulnerabilities in industrial communication protocols, inadequate authentication practices, and outdated firmware to compromise critical processes [26]. It may cause catastrophic effects, such as production line disruptions and equipment damage, and even pose a danger to human lives. The advanced nature of adversary tactics is in a state of continuous flux, making conventional intrusion detection mechanisms increasingly unsuitable [40].

Traditional IDS methods, such as signature- and anomaly-based approaches, remain core but are severely constrained in the modern era. Signature-based IDS relies on prior attack signatures, which are limited in their ability to identify zero-day and dynamic attacks.

Anomaly-based products aim to fill this gap by identifying unusual network activity, but they generate high false-positive warnings and swamp security teams with worthless alerts [5]. These limitations have created a need for more intelligent, responsive solutions capable of handling the volume and complexity of IIoT traffic.

Machine learning methods are another approach, more flexible at recognizing malicious patterns in network data. Unlike static rule-based systems, machine learning models continuously learn from data, enabling the detection of attacks that exploit subtle variations in traffic. By applying statistical learning to network behaviors, these models provide stronger generalization against unknown threats [7]. However, conventional machine learning methods still require extensive feature engineering and lack scalability for managing large volumes of IIoT data [9].

Deep learning has advanced this area by automatically learning features from large datasets and recognizing sophisticated patterns that standard models may not handle. Convolutional Neural Networks are capable of recognizing spatial relationships in organized traffic, while Multilayer Perceptrons perform well with tabular data to discriminate between usual and unusual activity [15]. These architectures have shown superior accuracy in intrusion detection tasks, making them attractive candidates for deployment in IIoT environments [44]. Furthermore, deep learning systems demonstrate adaptability to both known and emerging threats, positioning them as critical tools for modern industrial cybersecurity [12].

Centralized learning frameworks for IDS offer particular advantages over distributed approaches. Through merging data into a unified model, centralized systems support end-to-end monitoring of network flows, leading to more accurate and consistent intrusion detection. Federated learning, while helpful for ensuring privacy, faces synchronization delays, data heterogeneity, and high communication overhead that limit its real-world feasibility in large-scale IIoT scenarios [25], [60]. Centralized deep learning thus represents a more feasible and scalable option for securing industrial networks [10].

While significant progress has been made, comparisons across various deep learning models and frameworks for IIoT security are rare. Most of the current literature focuses on using a single model or framework without determining its relative performance in terms of accuracy, scalability, and in-the-field deployment. To address this gap, the present study evaluates the performance of CNNs and MLPs implemented in PyTorch and TensorFlow using the CIC IoT-DIAD 2024 dataset. The dataset provides diverse representations of industrial traffic and attack vectors, making it suitable for benchmarking [17].
The comparative evaluation in this study focuses on accuracy, precision, recall, F1-score, and confusion matrices as performance metrics. This analysis identifies the most effective architecture for real-time intrusion detection in IIoT-based critical infrastructure. The outcomes provide practical insights for enhancing industrial cybersecurity, offering a framework that improves detection rates, reduces false positives, and strengthens the resilience of industrial operations against increasingly sophisticated cyberattacks [11], [43].

2. RESEARCH AIM AND OBJECTIVES

The primary aim of this study is to design and evaluate a centralized learning-based intrusion detection system for Industrial IoT (IIoT) environments by applying and comparing deep learning models. To achieve this aim, the following objectives have been formulated:

1. To conduct a comprehensive review of deep learning approaches applied to intrusion detection in IIoT networks, with emphasis on their strengths, limitations, and suitability for industrial applications.
2. To preprocess the CIC IoT-DIAD 2024 dataset by addressing missing values, class imbalances, and feature scaling, thereby creating an optimized dataset for deep learning model training.
3. To implement four distinct deep learning models —PyTorch CNN, PyTorch MLP, TensorFlow CNN, and TensorFlow MLP —for detecting intrusions and anomalies in IIoT network traffic.
4. To evaluate and compare the performance of the implemented models using key metrics, including recall, precision, accuracy, F1-score, and a confusion matrix, to determine their detection effectiveness.
5. To analyse and interpret the comparative outcomes of the models, identify the most effective deep learning architecture for IIoT intrusion detection, and propose practical strategies for deployment in real-world industrial environments.

3. LITERATURE REVIEW

The widespread adoption of Industrial IoT (IIoT) has ushered in the promise of automation, efficiency, and connectivity, but it has also created significant security vulnerabilities. This has led to greater emphasis on intrusion detection systems (IDSs), with the inclusion of machine learning (ML) and deep learning (DL) algorithms to enhance resilience against dynamic cyber threats [62]. Various research contributions have explored centralized, distributed, and federated learning-based IDS frameworks to improve detection performance and adaptability in IIoT environments.

Elnakib et al. proposed the EIDM framework, which incorporates five deep learning models for malicious behaviour classification. The approach was tested on the CICIDS2017 dataset, achieving 99.48% accuracy for six-class classification and 95% accuracy in the complete 15-class setting. Unlike many previous studies, EIDM successfully classified all attack categories without merging similar classes, thereby demonstrating enhanced granularity [23]. However, the framework's lack of evaluation under real-time IIoT traffic conditions limits its demonstrated applicability.

Another contribution is the SS-Deep-ID model introduced by Abdel-Basset et al., which employed a semi-supervised spatio-temporal approach for IDS in fog-enabled IoT networks. Evaluations on CICIDS2017 and CICIDS2018 datasets showed 99% accuracy for binary classification and F1-scores between 75% and 99.85% for multiclass tasks [1].

While this semi-supervised design improves adaptability to unlabeled data streams, the computational overhead of CNN-based architectures poses a challenge for deployment in resource-constrained industrial environments.
The study by Toupas et al. developed a deep learning model with multiple hidden layers, trained on the CICIDS2017 dataset, for multiclass attack detection. This architecture achieved 99.95% accuracy, highlighting the capability of deep neural networks to handle diverse intrusion scenarios [65]. Despite this, the model risked overfitting because its evaluation relied on a single dataset, limiting its robustness to unseen industrial traffic. Similarly, Ullah and Mahmoud presented a deep learning-based anomaly detection model that generalizes across IoT networks, achieving promising accuracy but with computational efficiency challenges at scale [66].

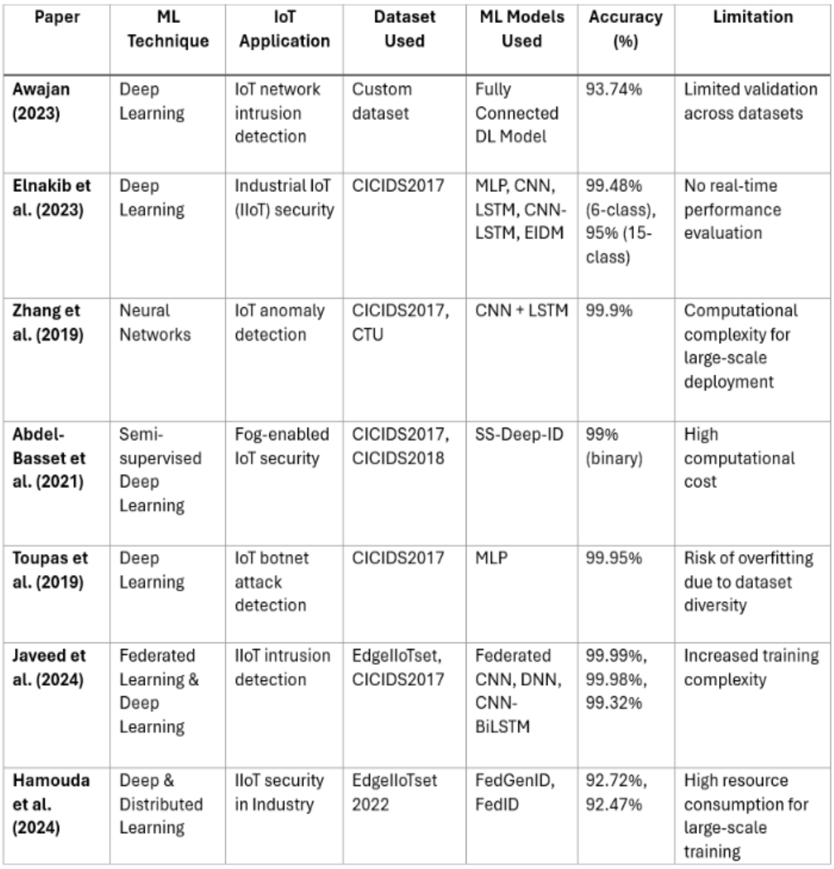
In the context of federated learning, Danish Javeed et al. introduced a zero-trust IDS framework utilizing CNN, DNN, and CNN-BiLSTM models. Trained on EdgeIIoTset 2022 and CICIDS2017, the models achieved near-perfect detection rates, with CNN and DNN models surpassing 99.98% accuracy. The federated setup preserved data privacy by avoiding centralized data pooling, though it introduced greater training complexity and resource demands [18]. Complementary to this, Hamouda et al. proposed FedGenID, a generative adversarial network (GAN)-driven federated IDS that improved detection of zero-day attacks by 10% over baseline models, but at the expense of higher computational resource consumption [27].

Other studies highlight hybrid IDS frameworks that integrate CNNs with recurrent neural architectures. For example, Zhang et al. designed CNN-LSTM models that were evaluated on the CICIDS2017 and CTU datasets, achieving an accuracy of up to 99.9% [20]. These models demonstrated the ability to capture both spatial and temporal features in IIoT traffic, although the scalability and real-time computational requirements remain concerns. Similarly, Wang et al. developed a CNN-BiLSTM model for IIoT intrusion detection that improved detection robustness across multiple attack types. However, its reliance on complex architectures raised deployment issues in edge environments [67].

In addition, Abdelaziz Testas explored classification models in PyTorch and TensorFlow, demonstrating the feasibility of integrating such frameworks into intrusion detection pipelines. These models demonstrated high performance across multiple datasets and highlighted key distinctions between backends, particularly in terms of training stability and scalability [2], [3]. Nonetheless, the lack of industrial-grade testing leaves questions about deployment readiness.

These works, by implication, all contribute to the increasing prevalence of deep learning in enhancing IDS development for IIoT security. Although detection accuracy has been established by models such as CNNs, LSTMs, BiLSTMs, and GANs on benchmark datasets, challenges remain. Such limitations include computational burden, limited dataset diversity, vulnerability to overfitting, and insufficient validation in real-world IIoT setups. Federated learning methods ensure privacy while providing solutions, but they add cost and increased convergence complexity [61]. Optimising DL architectures for lightweight, real-time detection and testing models on heterogeneous industrial datasets to make them generalisable are solutions to these issues.

Table 1: Literature review table summary



Paper	ML Technique	IoT Application	Dataset Used	ML Models Used	Accuracy (%)	Limitation
Awajan (2023)	Deep Learning	IoT network intrusion detection	Custom dataset	Fully Connected DL Model	93.74%	Limited validation across datasets
Elnakib et al. (2023)	Deep Learning	Industrial IoT (IIoT) security	CICIDS2017	MLP, CNN, LSTM, CNN-LSTM, EIDM	99.48% (6-class), 95% (15-class)	No real-time performance evaluation
Zhang et al. (2019)	Neural Networks	IoT anomaly detection	CICIDS2017, CTU	CNN + LSTM	99.9%	Computational complexity for large-scale deployment
Abdel-Basset et al. (2021)	Semi-supervised Deep Learning	Fog-enabled IoT security	CICIDS2017, CICIDS2018	SS-Deep-ID	99% (binary)	High computational cost
Toupas et al. (2019)	Deep Learning	IoT botnet attack detection	CICIDS2017	MLP	99.95%	Risk of overfitting due to dataset diversity
Javeed et al. (2024)	Federated Learning & Deep Learning	IIoT intrusion detection	EdgeIIoTset, CICIDS2017	Federated CNN, DNN, CNN-BiLSTM	99.99%, 99.98%, 99.32%	Increased training complexity
Hamouda et al. (2024)	Deep & Distributed Learning	IIoT security in Industry	EdgeIIoTset 2022	FedGenID, FedID	92.72%, 92.47%	High resource consumption for large-scale training

4. RESEARCH GAP

Although deep learning-based IDS solutions for IIoT have advanced, several limitations still hinder their deployment in the real world. Existing studies, including fully connected models [11], hybrid CNN-LSTM architectures [37], and anomaly-based detection frameworks [23], often lack systematic comparisons across different deep learning approaches. This makes it challenging to identify the most effective models for IIoT intrusion detection. Computational efficiency is another concern, as most models are not yet suitable for real-time recognition in large-scale environments [59]. Moreover, considerable emphasis has not been placed on comparing frameworks such as PyTorch and TensorFlow, leaving their respective strengths and weaknesses unaddressed.

Model robustness against adversarial attacks also represents a critical gap. While some solutions achieve high accuracy under controlled conditions, their adaptability to evolving threats remains largely untested. For instance, the EIDM model reached 95% accuracy on CICIDS2017 [23], and a CNN-LSTM approach reported 100% accuracy in binary tasks [37], but neither was evaluated against adaptive, real-world intrusions. Additionally, comparisons between architectures, such as CNNs for spatial feature extraction versus MLPs for structured traffic data, are limited, restricting insights into which provides the best trade-off between accuracy, efficiency, and scalability in IIoT contexts.

This research addresses these gaps by implementing and evaluating four deep learning models: PyTorch CNN, PyTorch MLP, TensorFlow CNN, and TensorFlow MLP. Using the CIC IoT-DIAD-2024 dataset, these models will be assessed through accuracy, recall, precision, F1-score, and confusion matrix. The study emphasizes centralized learning to enhance scalability and consistency, aiming to develop an IDS that balances high detection rates with practical efficiency and deployment feasibility.

5. METHODOLOGY

This study adopts an experimental methodology for the design, implementation, and evaluation of deep learning-based intrusion detection systems (IDS) tailored for Industrial IoT (IIoT) environments. The CIC IoT-DIAD 2024 dataset serves as the foundation for training and testing, providing heterogeneous network traffic that reflects real-world industrial scenarios with both benign and malicious flows. Four deep learning models are used: PyTorch CNN, PyTorch MLP, TensorFlow CNN, and TensorFlow MLP. They are all trained, validated, and tested in controlled settings, with performance evaluated against benchmark metrics such as accuracy, precision, recall, F1-score, and confusion matrices. This approach applies a comparative analysis to identify the best-performing architecture based on detection accuracy, computational overhead, and real-time usability. Ethical considerations regarding data privacy, as well as adherence to cybersecurity laws, are also taken into account, and the research is carried out in line with industry and academic standards.

5.1 Research Design

This study employs an experimental research design to develop and evaluate deep learning-based intrusion detection systems (IDS) for Industrial IoT (IIoT) environments. The CIC IoT-DIAD 2024 dataset serves as the basis for training and testing, as it contains diverse network traffic records, including both benign and malicious flows representative of real-world industrial systems. Preprocessing includes handling class imbalance, normalizing features, and preparing datasets for better training of deep learning models.

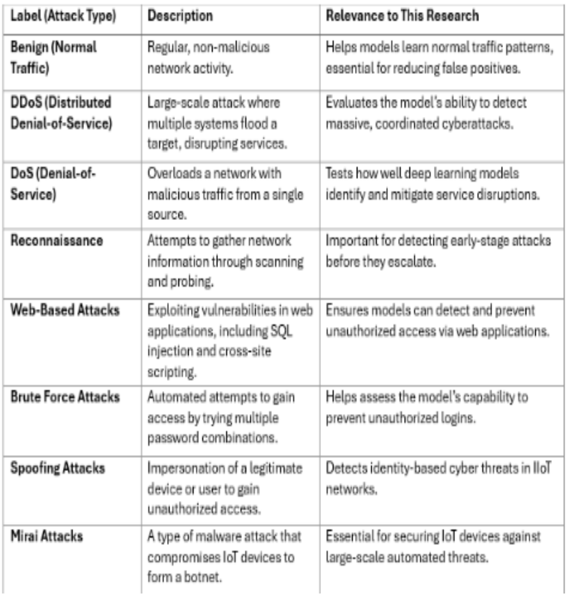
Two deep learning architectures, Multilayer Perceptrons (MLPs) and Convolutional Neural Networks (CNNs), are implemented in two leading frameworks, TensorFlow and PyTorch. This results in four experimental models: TensorFlow CNN, PyTorch CNN, TensorFlow MLP, and PyTorch MLP. CNNs are used to detect spatial correlations in traffic flows, while MLPs are applied to structured tabular data to identify non-linear feature interactions. They are trained, validated, and tested in controlled environments to ensure fair comparisons.

The assessment uses a quantitative method, with statistical and numerical calculations to quantify detection performance as numerical values. Accuracy, precision, recall, F1-score, and the confusion matrix are key performance indicators. The measures quantify the advantages and disadvantages of CNN and MLP architectures, as well as their PyTorch and TensorFlow implementations.

Modelling and validation are performed in Python using Jupyter Notebook, with the support of libraries such as Scikit-learn, Keras, TensorFlow, and PyTorch. Comparative analysis emphasizes differences in detection accuracy, computational cost, and scalability to identify the most practical and effective IDS framework for IIoT applications. The outcomes of this design are intended to guide the implementation of centralized IDS solutions that achieve both high detection rates and operational efficiency in industrial environments.

5.2 Dataset Description

The Canadian Institute produces the CIC IoT-DIAD 2024 dataset for Cybersecurity and is a large dataset that aims to advance IoT device identification research and anomaly detection [M. Rabbani et al., 2024]. It is filled with traffic from 105 IoT devices that were attacked by 33 types of attacks, grouped into seven categories: Web-based, Distributed Denial of Service (DDoS), Denial of Service (DoS), Spoofing (Man-in-the-Middle), Reconnaissance, Brute Force, and Mirai-based. These attacks were produced by compromised devices scanning other nodes within a network, simulating real-world IIoT security issues. The dataset is heterogeneous, combining packet-based and flow-based features to represent diverse aspects of traffic behaviour. A total of 73 behaviour-based features are included, including inter-arrival time, payload entropy, jitter, and handshake details, all of which are crucial for distinguishing benign traffic from malicious activity. Its dual suitability for both device identification and anomaly detection makes it highly relevant for evaluating deep learning-based intrusion detection systems. Table 2 explains the label attack types:



Label (Attack Type)	Description	Relevance to This Research
Benign (Normal Traffic)	Regular, non-malicious network activity.	Helps models learn normal traffic patterns, essential for reducing false positives.
DDoS (Distributed Denial-of-Service)	Large-scale attack where multiple systems flood a target, disrupting services.	Evaluates the model's ability to detect massive, coordinated cyberattacks.
DoS (Denial-of-Service)	Overloads a network with malicious traffic from a single source.	Tests how well deep learning models identify and mitigate service disruptions.
Reconnaissance	Attempts to gather network information through scanning and probing.	Important for detecting early-stage attacks before they escalate.
Web-Based Attacks	Exploiting vulnerabilities in web applications, including SQL injection and cross-site scripting.	Ensures models can detect and prevent unauthorized access via web applications.
Brute Force Attacks	Automated attempts to gain access by trying multiple password combinations.	Helps assess the model's capability to prevent unauthorized logins.
Spoofing Attacks	Impersonation of a legitimate device or user to gain unauthorized access.	Detects identity-based cyber threats in IIoT networks.
Mirai Attacks	A type of malware attack that compromises IoT devices to form a botnet.	Essential for securing IoT devices against large-scale automated threats.

Table 2

For this study, a subset of five labels was selected: benign, DoS, DDoS, Spoofing (MiTM), and Mirai. These labels were chosen to capture common and impactful threats in IIoT environments. The four deep learning models implemented —PyTorch CNN, PyTorch MLP, TensorFlow CNN, and TensorFlow MLP —are trained to classify these categories, enabling the detection of both volumetric and stealth-based attacks. The high-quality labeling and diversity of attack vectors in CIC IoT-DIAD 2024 provide a strong foundation for building models capable of accurate, adaptive, and real-time intrusion detection in industrial IoT systems.

5.3 Data Extraction and Preprocessing

The CIC IoT-DIAD 2024 dataset was first loaded from CSV into a structured pandas DataFrame for analysis, with a random seed of 20 to ensure consistency across experimental runs. To counteract class imbalance, undersampling was used to reduce the majority class sizes while retaining all minority class instances, resulting in an equal sample ratio to prevent overfitting during model training.

The dataset was cleaned to ensure quality and consistency. Column names were normalized by removing leading and trailing whitespaces and substituting internal spaces with underscores to prevent reference errors when the code is executed. NaN and out-of-range values were treated carefully: 178 NaN values in column Flow_Bytes/s, 81 infinite values in Flow_Bytes/s, and 259 in Flow_Packets/s. Infinite values were substituted with NaN, and all rows that had NaN values were deleted to leave complete records in the dataset to train on. Redundant features were also eliminated, including seven single-valued columns with no variability, as they contributed no valuable information to classification tasks.

Additionally, 69 duplicate rows were detected and removed to prevent over-representation of specific traffic patterns, which could bias model predictions. Finally, irrelevant identifier-based columns that added noise without contributing predictive value were dropped, streamlining the dataset and reducing computational overhead. All these preprocessing operations, combined, ensured that the dataset was clean, representative, and optimized for intrusion detection modelling.

5.4 Exploratory Data Analysis

The initial dataset contained 192,787 rows and 84 columns, but following data cleaning, it was reduced to 192,459 rows and 73 columns. Except for one column, they are numeric, i.e., continuous (floats) or discrete (integers). The single categorical column contains qualitative labels used to classify observations into distinct traffic categories.

In this study, the categorical field Label is defined as the target variable, representing the output that the deep learning models are designed to predict. This field classifies IIoT network traffic into five categories: Benign, Distributed Denial of Service (DDoS), Denial of Service (DoS), Mirai, and Man-in-the-Middle (MITM). Four of these categories —DDoS, DoS, Mirai, and MITM —identify specific types of cyberattacks with distinct behaviors and threat profiles, whereas class Benign identifies legitimate IIoT network traffic that is not a security threat.

Since the aim of the study is to categorize IIoT traffic into one of these five classes, the task is approached as a multi-class classification problem. This enables the models not only to identify malicious traffic but also to distinguish among various types of intrusions. Proper classification improves IIoT security by enabling the appropriate identification of threats, timely detection, and effective countermeasures against security attacks in industries.

Figure 4 presents the top three source port numbers for each traffic category. The study indicates that ports 53 and 443 are consistently the most active in both benign and malicious traffic. Port 53 is associated with the Domain Name System (DNS), which converts domain names into IP addresses and is crucial to network communication. Port 443 is used for Hypertext Transfer Protocol

Secure (HTTPS), which provides secure, encrypted data transfer. The prevalence of these ports across categories highlights their use in daily operations, but also shows that attackers use them to conceal nefarious activity within standard traffic patterns.

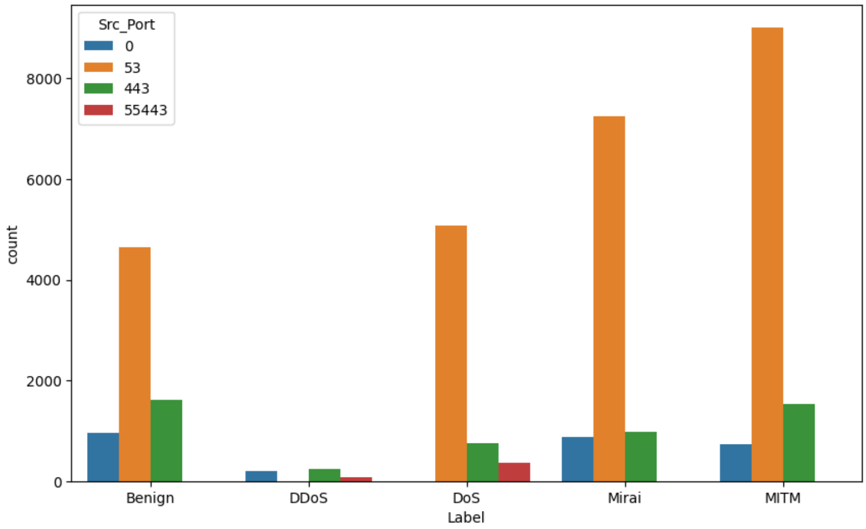


Figure 4: Top Three Source Ports across IIoT Traffic Categories

Figure 5 displays the three most frequently used destination ports for each traffic category in the Industrial IoT network. In all categories, ports 53 and 443 appear as the top destination ports. The two are significant for network communication because port 53 handles DNS services, while port 443 handles HTTPS traffic. The fact that these ports appear in various categories means that both malicious and legitimate traffic will use them to facilitate communication and exchange data.

Conversely, Distributed Denial of Service (DDoS) attacks exhibit a distinct pattern, as they primarily scan port 55442. The port is not commonly used for ordinary network services, and its use in DDoS traffic indicates that attackers are exploiting flaws or targeting services running on this port to conduct massive, disruptive attacks.

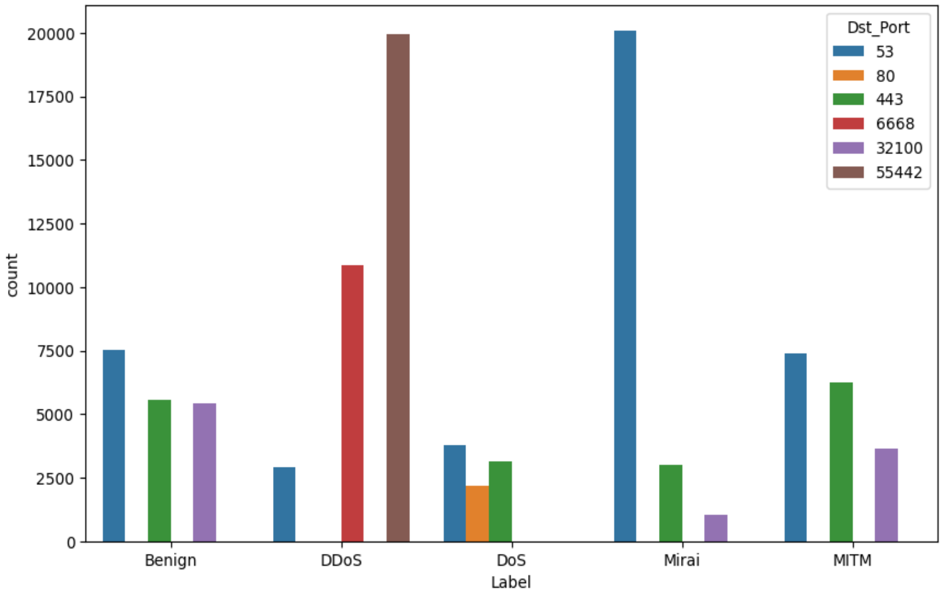


Figure 5: Top Three Destination Ports across IIoT Traffic Categories

Figure 6 shows the distribution of protocol types in the data set, with Protocol 17 —the User Datagram Protocol (UDP) —having the highest prevalence. UDP is a lean, connectionless protocol that enables quick, low-latency data transfer without requiring a dedicated connection between machines. Its prevalence in the dataset indicates that most IIoT devices use UDP for low-latency data transport in time-critical applications such as real-time monitoring and industrial control.

Protocol 6, the Transmission Control Protocol (TCP), is the second most prevalent protocol in the dataset. TCP is connection-oriented, unlike UDP, and provides reliable, ordered data transport by establishing a secure connection before data transfer. The widespread use of TCP in the dataset indicates its significance in IIoT networks, particularly in high-reliability applications such as firmware updates, device management, and systematic data logging.

This protocol distribution reflects the dual communication requirements in Industrial IoT networks: UDP provides fast, lightweight communication, while TCP provides secure, reliable data transfer.

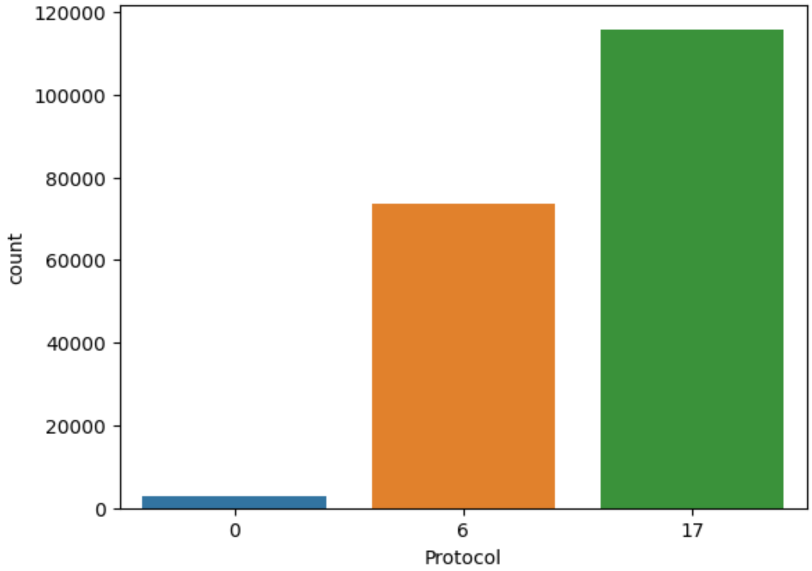


Figure 6: Distribution of Protocol Types in IIoT Traffic

Figure 7 shows the mean flow duration for each protocol type. Protocol 17 (UDP) has the shortest mean flow duration, consistent with its connectionless nature, enabling rapid, efficient data transport without the expense of establishing or maintaining long-lasting connections. Protocol 6 (TCP) maintains a longer mean flow duration, consistent with its connection-oriented approach, which guarantees complete, in-order delivery by maintaining constant contact with the sender and receiver.

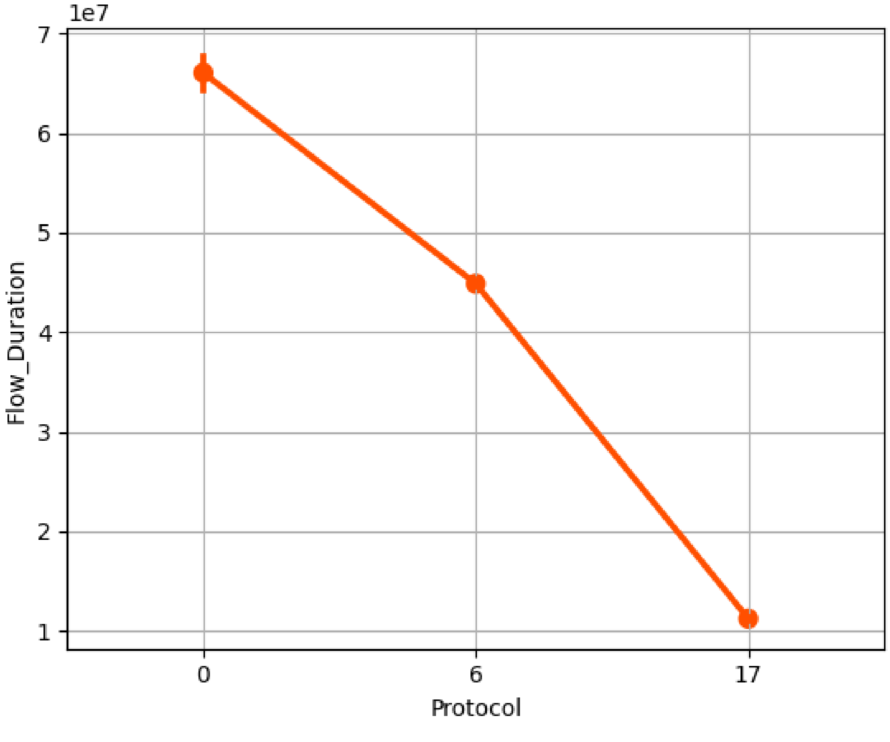


Figure 7: Average Flow Duration of Protocol Types

Figure 8 depicts the maximum Down/Up Ratio across different categories of IIoT network traffic. The Down/Up Ratio is the ratio of downloaded (inbound) to uploaded (outbound) traffic in a network flow. In the Benign class, it reaches 10, where there is a relatively balanced exchange in which downloads exceed uploads but remain within a healthy range. That's characteristic of industrial IoT settings, where devices receive many instructions, firmware updates, or sensor data and send only minimal acknowledgement packets or status reports.

Conversely, Mirai-type attacks exhibit a highly asymmetric pattern, with the maximum Down/Up Ratio exceeding 10,000. Mirai botnets generate enormous amounts of outbound malicious traffic with minimal inbound response, creating extreme data-flow asymmetry. This unusually high ratio is a strong indicator of device compromise and can serve as a significant feature for intrusion detection in IIoT security frameworks.

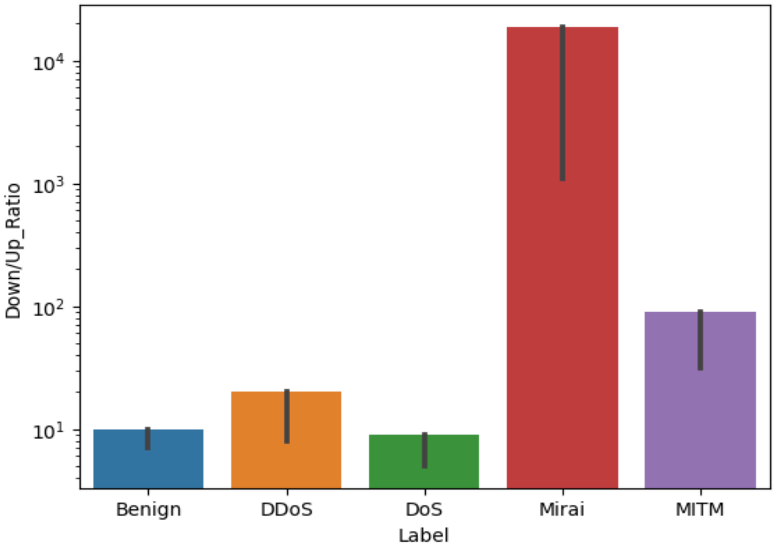


Figure 8: Maximum Down/Up Ratio in IIoT Traffic Categories

The distribution of forward packets across IIoT network traffic categories is illustrated in Figure 9. Among the five classes, the DoS attack class shows the greatest variation, indicating significant fluctuations in the number of packets forwarded in this class. This reflects the malleability of DoS attacks, which can range from low-level to large-scale packet forwarding, depending on the attack strategy employed.

Additionally, the DoS class has the highest total number of forward packets, reflecting its tendency to produce the most packets overall. This feature aligns with the overall objective of DoS attacks: to overwhelm target systems with a large volume of traffic.

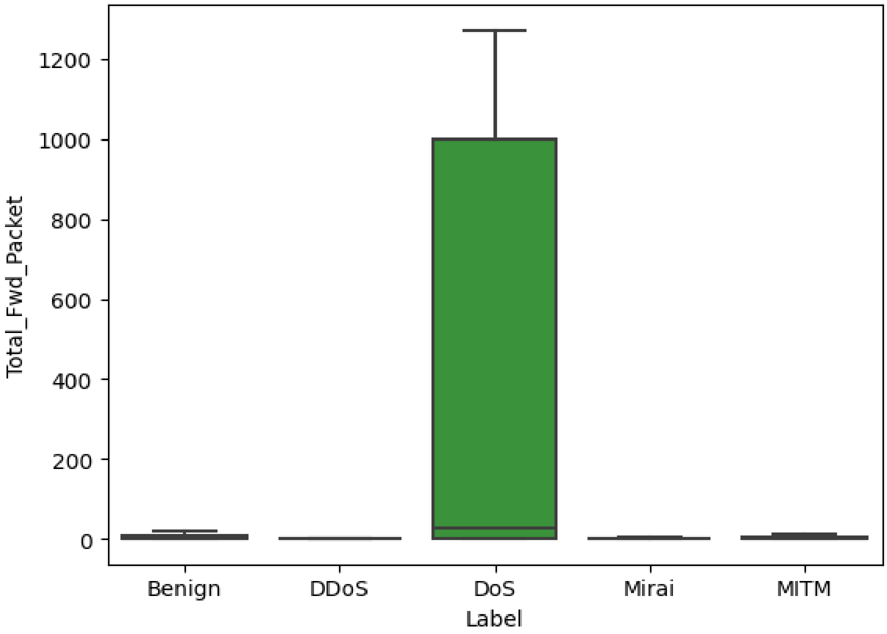


Figure 9: Distribution of Forward Packets in IIoT Traffic Categories

The distribution of backward packets across different IIoT network traffic classes is presented in Figure 10. Compared to forward packets, the overall range of backward packets is notably smaller, showing that backward communication is generally more limited in both volume and variability.

Among the five categories, the Benign class records the widest range and the largest number of backward packets, reflecting that normal IIoT traffic typically involves richer bidirectional communication. In contrast, attack-oriented categories display narrower ranges and fewer backward packets. This is best described by the DDoS class, which has the smallest range and the lowest total of backward packets. Such low reverse traffic is characteristic of DDoS attacks, which aim to overwhelm a target with heavy forward traffic, leaving little room for legitimate responses. Such deviations in the reverse packet distribution underscore critical behavioral differences between benign and malicious traffic, offering helpful pointers toward intrusion detection for IIoT systems.

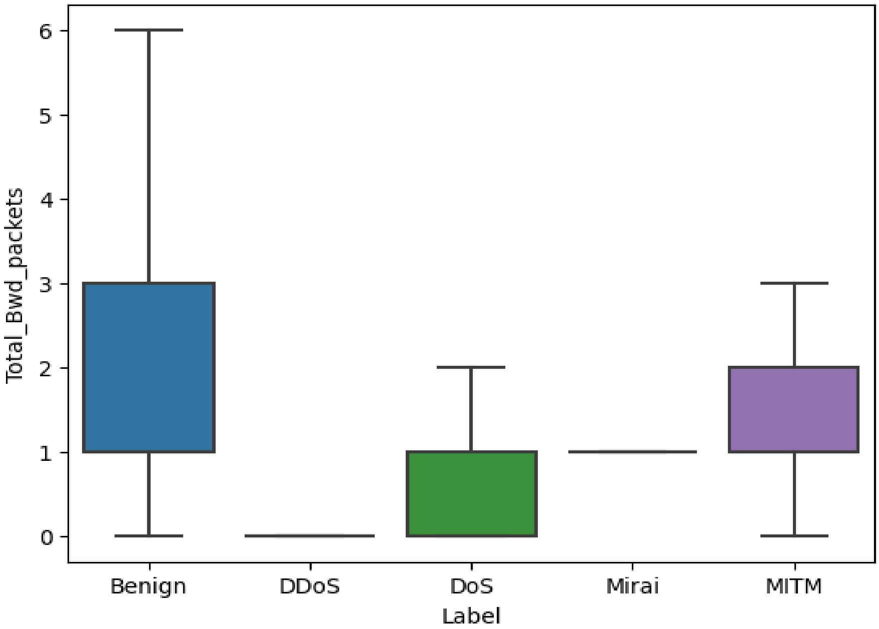


Figure 10: Distribution of Backward Packets in IIoT Traffic Categories

Figure 11 shows the mean packet size for each class of Industrial IoT network traffic. The DDoS class has the highest mean packet size, which is significantly higher than the other classes. DDoS class packets are about 15 times the size of Benign class packets on average, further underscoring the intrusive nature of DDoS traffic compared to normal IIoT communication.

This significant variance in packet size reflects the strategy of DDoS attacks, which send many very large packets to flood and weaken targeted systems. Benign traffic, in contrast, refers to legitimate IIoT communication that typically consists of smaller, consistent packet sizes. The sharp distinction between attack and normal traffic emphasizes the importance of packet-size monitoring as a critical feature for intrusion detection in IIoT security systems.

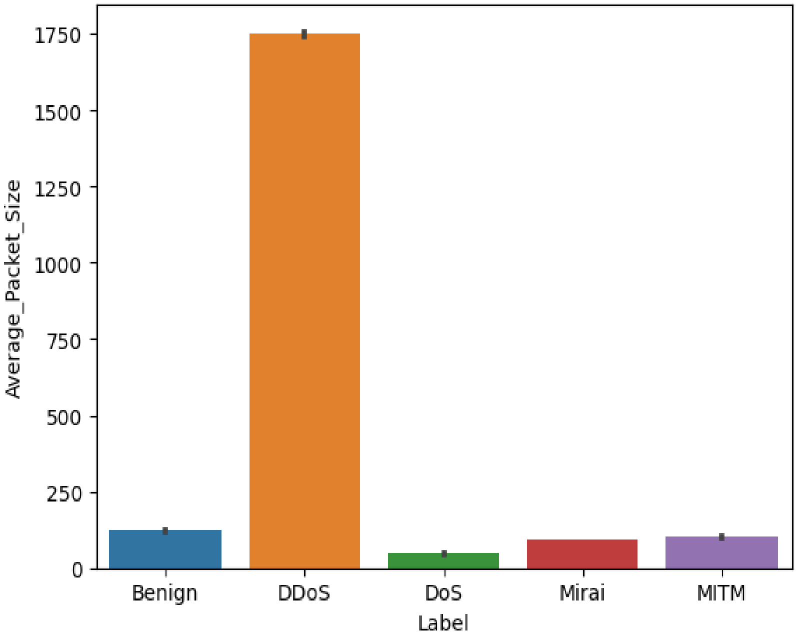


Figure 11: Average Packet Size across IIoT Traffic Categories

Figure 12 illustrates the average number of ACK (Acknowledgement) flags across various IIoT network traffic classes and the use of acknowledgement messages in different communication patterns. The maximum ACK flag numbers are achieved in the Benign class, indicating frequent usage of acknowledgements in typical IIoT communications for effective data exchange and device communication reliability.

However, the DDoS, DoS, and Mirai attack classes have significantly smaller ACK flag values, reflecting the fact that these kinds of malicious traffic do not use acknowledgement mechanisms. This is consistent with the purposeful destructiveness of such attacks, which aim to flood or destabilize network services over constant, acknowledged communication. The strong separation between Benign and attack traffic highlights the value of ACK flag counts as a discriminative intrusion-detection metric for IIoT environments.

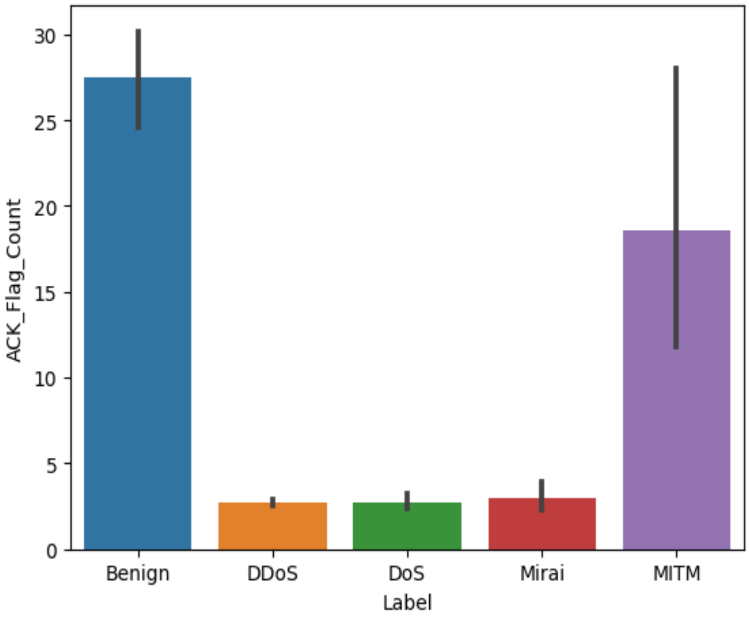


Figure 12: Average ACK Flag Count across IIoT Traffic Categories

Figure 13 displays the average length of forward packets across different classes of IIoT network traffic, where the forward packet length refers to the size of data packets transmitted in the forward direction within a network session. The outcomes indicate significant differences in large-packet-size patterns between malicious and benign.

The DDoS class indicates the highest mean forward packet length, as expected in Distributed Denial of Service attacks that send large volumes of packets to overload target systems and achieve maximum network saturation. Benign traffic, in contrast, indicates relatively small, consistent forward packet sizes typical of normal communication streams. This disparity underscores the importance of forward packet length as a feature for distinguishing between regular IIoT traffic and disruptive cyberattacks.

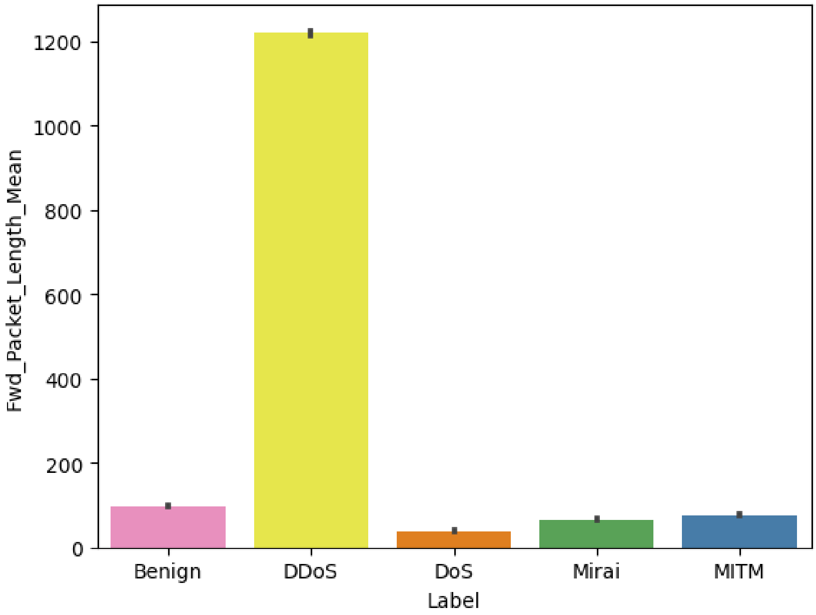


Figure 13: Average Forward Packet Length across IIoT Traffic Categories

Figure 14 illustrates the average length of backward packets across different classes of IIoT network traffic. The backward packet length represents the size of response packets sent after receiving data in a network session. The results indicate clear distinctions between benign and attack-related traffic.

The Benign category records the highest average backward packet length, reflecting the larger response packets typical of normal, legitimate communication in IIoT environments. This is closely followed by the MITM class, where intercepted and manipulated communications also produce substantial backward packet sizes.

Conversely, the DDoS and DoS classes have the smallest backward packet sizes, a reference to the nature of such attacks, which emphasize overwhelming targets with enormous requests rather than engaging in substantial bidirectional communication. These differences highlight the relevance of backward packet length as a feature for distinguishing benign activity from malicious intrusions.

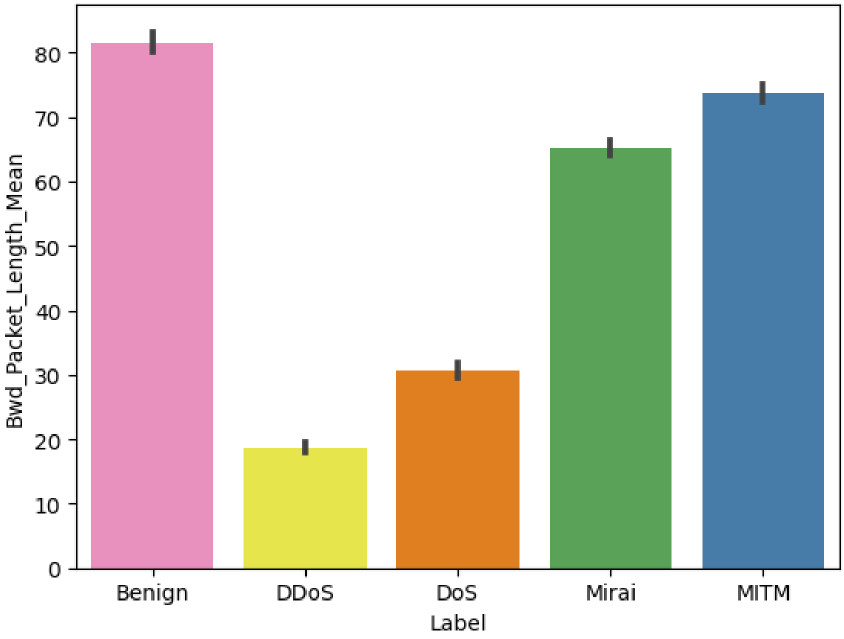


Figure 14: Average Backward Packet Length across IIoT Traffic Categories

5.5 Feature Engineering

The dataset used in this study is predominantly numerical, which suits machine learning models since deep learning algorithms can directly process numeric attributes. The target variable (Label), however, is categorical and was converted into numeric codes: Benign (0), DDoS (1), DoS (2), MITM (3), and Mirai (4). This encoding preserved class distinctions while making the data compatible with deep learning models.

For framework compatibility, labels were handled differently. In TensorFlow, labels were one-hot encoded to match the requirements of categorical_crossentropy, while in PyTorch, integer class indices were retained for use with CrossEntropyLoss. The predictor variables were separated from the target variable, and the dataset was split into 80% for training and 20% for testing. To ensure fair learning, all features were normalized using scaling parameters fit only on the training set, avoiding data leakage.

For CNN models, an additional dimension was added to the input features to match the expected multi-dimensional structure, enabling the extraction of spatial and structural patterns. Figure 15 illustrates the distribution of traffic categories, showing a relatively balanced dataset that reduces the risk of bias and supports effective classification across all classes.

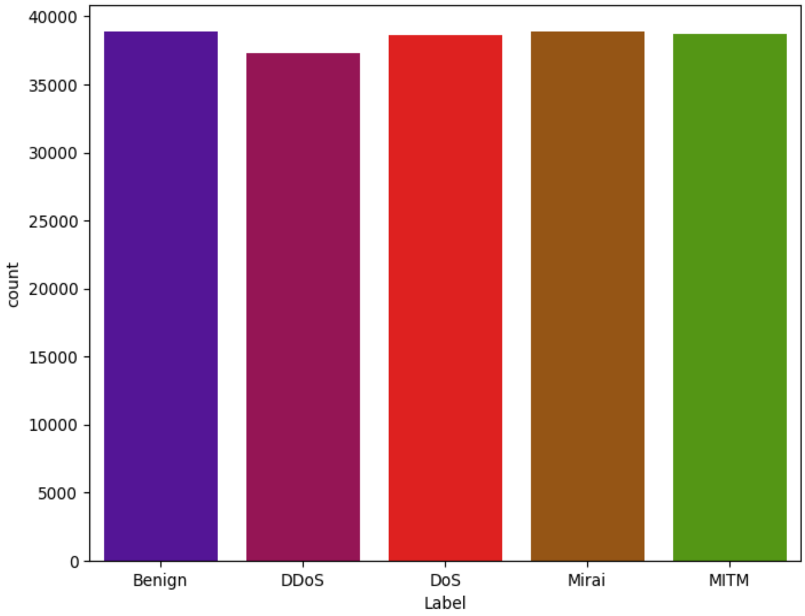


Figure 15: IIoT Network Traffic Categories

6. MODEL DEVELOPMENT

This study developed two deep learning models, a Multi-Layer Perceptron (MLP) and a Convolutional Neural Network (CNN), implemented in both TensorFlow and PyTorch, yielding four variants: TensorFlow MLP, PyTorch MLP, TensorFlow CNN, and PyTorch CNN. The classification task was multi-class, enabling the detection of multiple IIoT attack categories.

The MLP architecture included an input layer, four hidden layers with ReLU activation, batch normalization after the second layer, and dropout after the third to prevent overfitting. A SoftMax output layer was used for probability-based classification. The CNN model consisted of two 1D convolutional layers with ReLU activation, MaxPooling1D for dimensionality reduction, and flattening, followed by three fully connected layers. Batch normalization was applied before the final SoftMax output layer.

All models were trained on the same dataset splits, using similar optimizers, to ensure a fair comparison and reliable performance evaluation.

6.1 Model Evaluation

After training, models were tested on a completely different dataset they had not seen during training. This would ensure that the outcome was based on generalization ability rather than memorization. The assessment used a set of measures, including accuracy, precision, recall, F1-score, and a confusion matrix, to provide a comprehensive picture of model performance in classifying normal vs. malicious Industrial IoT network traffic. The above parameters are commonly used in machine learning studies because they can estimate various characteristics of classification performance, such as overall accuracy, minority class detection, and the balance between false positives and false negatives.

6.2 Accuracy of the Various Models Employed

Figure 16 shows the accuracy levels achieved by the various models. Of the four, the highest accuracy was achieved by the TensorFlow CNN model, which reached 80.1%, making it the best model for intrusion detection in this research. This discovery has significant implications for U.S. organizations that rely on efficient real-time surveillance of IIoT networks, such as those operating power distribution networks or computerized manufacturing plants. The PyTorch CNN closely followed with 77.3% accuracy, confirming the robustness of convolutional architectures in extracting complex patterns from IIoT traffic data. The TensorFlow MLP achieved an accuracy of 75.6%, which is below par, and the PyTorch MLP achieved the lowest accuracy of 68.9%.

The comparative outcome suggests that convolutional neural networks are more suitable to intrusion detection in U.S. organizational IIoT-based critical infrastructure because they can detect finer structural patterns and network traffic interdependencies more effectively than MLPs. This is critical in industries like U.S. energy and transportation, where intrusions detected proactively can be the difference between ongoing operations and complete disruption.

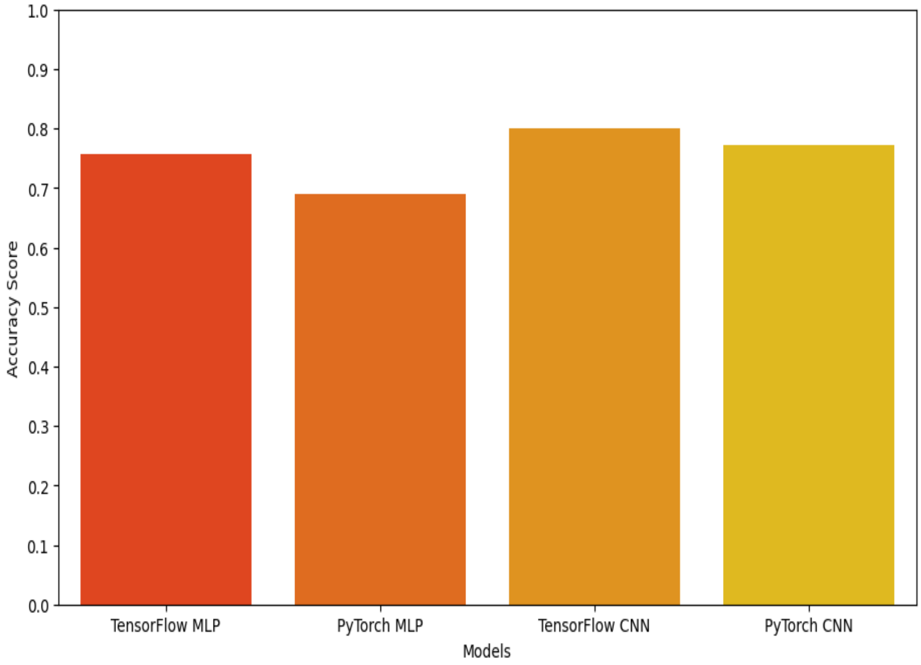
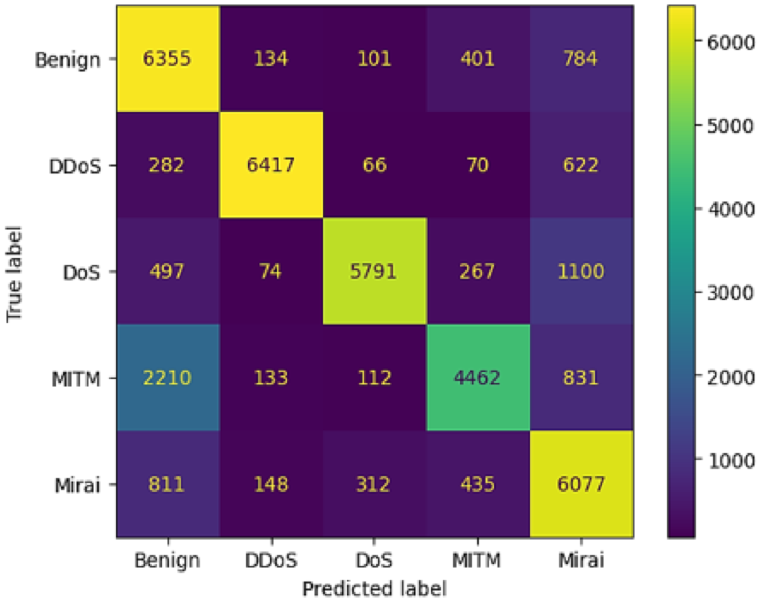


Figure 16: Accuracy Scores of Models

6.3 Confusion Matrix

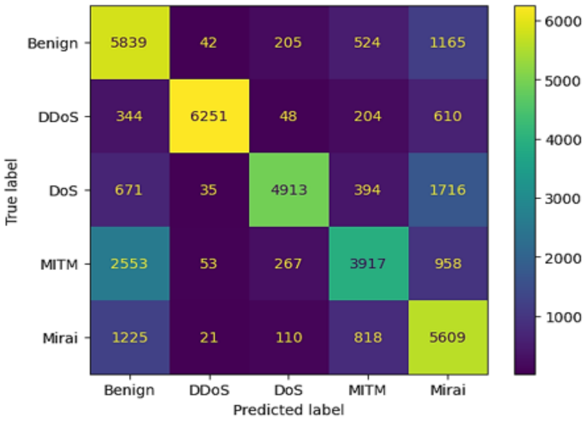
The confusion matrices for the four trained models are presented in Figures 17(a–d) and provide a comprehensive overview of classification outcomes. A confusion matrix presents the counts of correct and incorrect classifications across all classes and provides a clearer understanding of how well each model distinguishes between benign and malicious Industrial IoT (IIoT) network traffic. This type of assessment is particularly important in predictive analytics, while upholding U.S. organizations in industries like energy, manufacturing, and healthcare, where misclassification errors can lead to unexpected violations and widespread disruptions of critical infrastructure.

The **TensorFlow MLP** model exhibited a high misclassification rate, with 1420 Benign traffic instances, 1040 DDoS instances, 1938 DoS instances, 1706 Mirai instances, and 3286 MITM instances misclassified. These results indicate that, although the model was successful in identifying patterns, its accuracy for true real-time deployment in U.S. critical infrastructure might not be sufficient to carry out vital detection that would help lower operational risks.



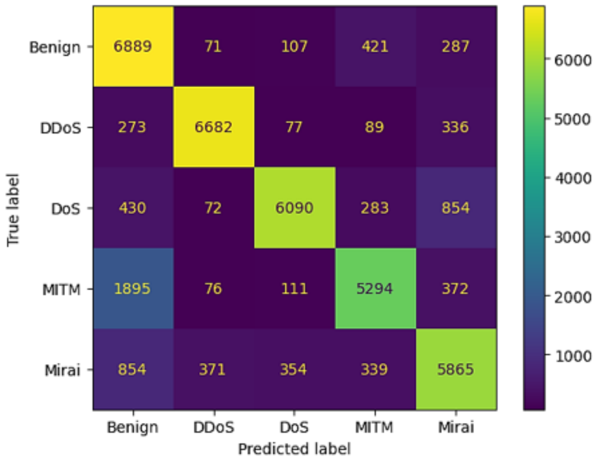
a) TensorFlow MLP

PyTorch MLP was less accurate, misclassifying 1936 Benign traffic examples, 1206 DDoS examples, 2816 DoS examples, 2174 Mirai examples, and 3831 MITM examples. The error counts are so high that even fully connected models cannot handle the complexity of IIoT traffic and therefore are not suitable for large-scale deployment within U.S. companies responsible for protecting sensitive operational networks.



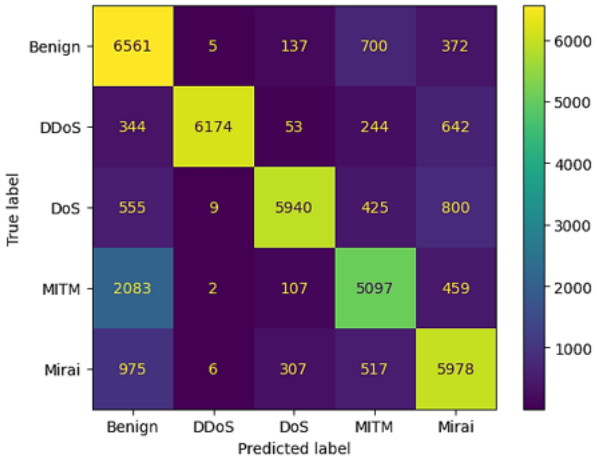
b) PyTorch MLP

The **TensorFlow CNN** showed stronger classification performance, misclassifying 886 Benign, 775 DDoS, 1639 DoS, 1918 Mirai, and 2454 MITM instances. These reduced misclassification counts highlight the CNN's ability to extract structural patterns from IIoT traffic, making it a stronger candidate for deployment in U.S. industrial organizations where accurate attack detection can safeguard operations such as automated manufacturing or power grid monitoring.



c) TensorFlow CNN

The **PyTorch CNN** also performed relatively well, misclassifying 1214 Benign, 1283 DDoS, 1789 DoS, 1805 Mirai, and 2651 MITM instances. While not as precise as the TensorFlow CNN, it still demonstrates that convolutional approaches outperform MLP-based methods for handling complex IIoT intrusion detection tasks.



d) PyTorch CNN

Overall, the confusion matrix report demonstrates that convolutional models, and in this case TensorFlow CNN, achieve better classification results than MLP. Pragmatically, this means American businesses that operate critical IIoT infrastructure benefit from implementing CNN-based models, as they can better reduce misclassification errors, identify threats in real time, and bolster defenses against cyberattacks on critical services.

6.4 Classification Report

Tables 12 to 15 provide a summary of the classification report for all models and indicate their accuracy in correctly classifying instances of each intrusion category. It also provides key performance measures, such as precision, recall, and F1-score.

Table 12: Classification Report of TensorFlow MLP

Category	Precision	Recall	F1-score
Benign	0.63	0.82	0.71
DDoS	0.93	0.86	0.89
DoS	0.91	0.75	0.82
Mirai	0.65	0.78	0.71
MITM	0.79	0.58	0.67

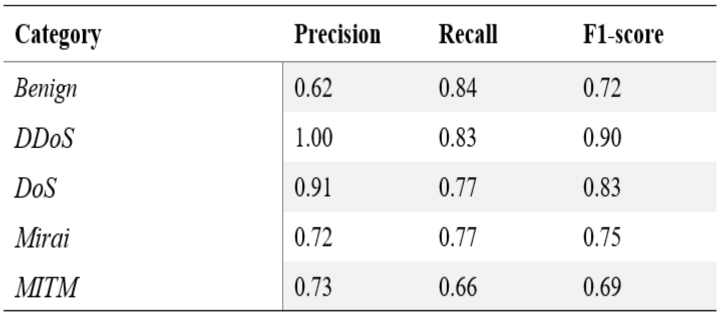
Table 13: Classification Report of PyTorch MLP

Category	Precision	Recall	F1-score
Benign	0.55	0.75	0.63
DDoS	0.98	0.84	0.90
DoS	0.89	0.64	0.74
Mirai	0.56	0.72	0.63
MITM	0.67	0.51	0.58

Table 14: Classification Report of TensorFlow CNN

Category	Precision	Recall	F1-score
Benign	0.67	0.89	0.76
DDoS	0.92	0.90	0.91
DoS	0.90	0.79	0.84
Mirai	0.76	0.75	0.76
MITM	0.82	0.68	0.75

Table 15: Classification Report of PyTorch CNN



Category	Precision	Recall	F1-score
Benign	0.62	0.84	0.72
DDoS	1.00	0.83	0.90
DoS	0.91	0.77	0.83
Mirai	0.72	0.77	0.75
MITM	0.73	0.66	0.69

The provided tables show that the TensorFlow MLP performed extremely well at identifying DDoS and DoS attacks, achieving high precision and recall. This implies that in the case of an American critical infrastructure, for example, power networks or water purification plants, the model would be useful in the detection of wholesale denial-of-service disruption before its ability to cripple services. But it worked quite well in detecting Benign traffic with a false positive bias, which might overwhelm security personnel in U.S. organizations with meaningless alarms. Its MITM and Mirai attack detection was weak, suggesting a lack of ability to detect botnet-based intrusions or hidden interception attempts, both of which are very germane to U.S. smart manufacturing networks and networked healthcare settings.

PyTorch MLP excelled in detecting DDoS attacks, with a high F1-score, showing its capability to accurately catch volumetric attacks that threaten financial institutions and communication networks in the U.S. It showed decent performance for Mirai attacks but struggled with precision, meaning it could mislabel safe traffic as malicious, a problem for operational environments like oil and gas plants where downtime is costly. In the Benign category, it had moderate recall but low precision, leading to legitimate activities being flagged. For DoS attacks, it had high precision but lower recall, suggesting that some attack traffic might pass undetected, which could be critical in emergency response networks. Its weakest performance was in identifying MITM attacks, with the lowest recall, indicating difficulty in protecting against stealthy traffic interception —a serious concern for U.S. defense and aerospace organizations where data confidentiality is vital.

The TensorFlow CNN model performed well across all categories. It showed strong performance in detecting Benign instances and various types of intrusions, with good precision and recall, resulting in balanced F1 Scores. In practice, this balance would be valuable to U.S. organizations such as utility providers and logistics companies, where distinguishing normal industrial IoT communication from real threats is key to maintaining operational continuity while avoiding alert fatigue.

The PyTorch CNN model demonstrated strong performance in detecting DDoS attacks, with high precision and recall, making it suitable for U.S. cloud-based IIoT deployments vulnerable to traffic flooding. It showed good recall for Benign instances but lower precision, leading to more false positives, which could drain resources in large-scale industrial control centers. For DoS attacks, the model was exact but had lower recall, indicating that while it may correctly identify some attack traffic, other attacks may go unnoticed —a limitation that could impact smart grid operators. It performed reasonably well for Mirai attacks, with balanced precision and recall, which is critical for U.S. healthcare IoT networks increasingly targeted by botnets. Its weakest performance was in detecting MITM attacks, where it had the least recall, exposing a vulnerability in protecting against advanced persistent threats targeting critical communication channels in U.S. government systems.

7. MODEL RESULT IMPLICATIONS FOR U.S. CRITICAL INFRASTRUCTURE IDS

The ubiquity of Industrial IoT (IIoT) technologies in U.S. critical infrastructure verticals, such as energy, manufacturing, and transportation, has significantly improved operational productivity and enabled real-time decision-making. Although this added connectivity introduces intricate security challenges that attackers can exploit to compromise services [34], the results of this research indicate that deep learning intrusion detection systems, such as convolutional neural network (CNN) models, yield robust performance for identifying categories of cyber threats in IIoT traffic. For U.S. companies, this demonstrates the viability of integrating centralized learning-based IDS into industrial control systems for operational use to enhance resilience against ever-changing threats [58].

In practice, the deployment of platforms such as TensorFlow CNN, which performed the best in this study, can assist U.S. companies in detecting and neutralizing threats, such as DDoS and DoS attacks, before they cause extensive service outages. This is especially true for organizations that operate critical infrastructure, where failure or data tampering will have major economic and safety implications [62]. In addition, the capacity of these models to process traffic in real time further enhances their active monitoring capabilities, aligning with the cybersecurity standards established by U.S. regulators to protect critical systems.

8 RESEARCH CONCLUSION

This research demonstrates that deep learning models are highly efficient for IIoT network intrusion detection, and that TensorFlow CNN achieved the best performance (80.1% accuracy), followed by TensorFlow MLP and PyTorch CNN. The experiments indicate that CNNs outperform in extracting hierarchical and spatial correlations in IIoT traffic, thus accurately detecting anomalies. Whereas all the models performed poorly under MITM attacks, the TensorFlow CNN had the best-balanced detection across classes, including DDoS, DoS, and Mirai. Such results highlight the significance of model choice and framework performance in real-world deployment of IDS in industry settings where reliability and timely response are essential to protecting critical infrastructure.

9. RESEARCH LIMITATIONS

Although successful, this work faces limitations that affect generalizability. First, it relied solely on the CIC IoT-DIAD 2024 dataset, which, while realistic, may not cover all possible IIoT scenarios. Second, hyperparameter tuning and optimization were not conducted, meaning models may not have reached their best performance, particularly for challenging classes such as MITM. Third, the study used centralized learning only, without exploring federated approaches, which could be beneficial in distributed IIoT environments where privacy and bandwidth are concerns. Finally, all models were tested on pre-processed data in a controlled environment, leaving their real-time adaptability to live industrial networks untested.

10. RESEARCH RECOMMENDATIONS/FUTURE WORK

Further research must employ strict validation methods, such as k-fold cross-validation and hyperparameter tuning, to improve performance across all classes, particularly for sophisticated attacks. Adaptive feature selection techniques and ensemble or hybrid approaches (e.g., CNN + MLP) may be used to improve detection accuracy at the cost of reduced computational expense. In practice, using the TensorFlow CNN within current industrial security systems, such as firewalls and IDS, would greatly enhance threat identification, particularly against DoS and DDoS attacks. Moreover, using diverse datasets and live IIoT traffic for testing, along with threat intelligence data, would ensure scalability, flexibility, and enhanced resistance to growing cyber threats in critical infrastructure networks.

REFERENCES

[1] M. Abdel-Basset, H. Hawash, R. K. Chakrabortty, and M. J. Ryan, “Semi-supervised Spatio-Temporal Deep Learning for Intrusions Detection in IoT Networks,” IEEE Internet of Things Journal, pp. 1–1, 2021, doi: https://doi.org/10.1109/jiot.2021.3060878.

[2] Abdelaziz Testas, “Deep Learning with PyTorch for Classification,” Apress eBooks, pp. 321–429, Jan. 2024, doi: https://doi.org/10.1007/979-8-8688-1017-6_6.

[3] Abdelaziz Testas, “Deep Learning with TensorFlow for Classification,” Apress eBooks, pp. 431–488, Jan. 2024, doi: https://doi.org/10.1007/979-8-8688-1017-6_7.

[4] G. Abbas, A. Mehmood, M. Carsten, G. Epiphaniou, and J. Lloret, “Safety, Security and Privacy in Machine Learning Based Internet of Things,” Journal of Sensor and Actuator Networks, vol. 11, no. 3, p. 38, Jul. 2022, doi: https://doi.org/10.3390/jsan11030038.

[5] N. Abosata, S. Al-Rubaye, and G. Inalhan, “Customised Intrusion Detection for an Industrial IoT Heterogeneous Network Based on Machine Learning Algorithms Called FTL-CID,” Sensors, vol. 23, no. 1, p. 321, Dec. 2022, doi: https://doi.org/10.3390/s23010321.

[6] H. Alavizadeh, J. Jang-Jaccard, and H. Alavizadeh, “Deep Q-Learning based Reinforcement Learning Approach for Network Intrusion Detection,” arXiv.org, 2021. https://arxiv.org/abs/2111.13978

[7] M. Al-Ambusaidi, Zhang Yinjun, Y. Muhammad, and A. Yahya, “ML-IDS: an efficient ML-enabled intrusion detection system for securing IoT networks and applications,” Soft Computing, Dec. 2023, doi: https://doi.org/10.1007/s00500-023-09452-7.

[8] F. Al-Quayed, Z. Ahmad, and M. Humayun, “A situation based predictive approach for cybersecurity intrusion detection and prevention using machine learning and deep learning algorithms in wireless sensor networks of industry 4.0,” IEEE access, pp. 1–1, Jan. 2024, doi: https://doi.org/10.1109/access.2024.3372187.

[9] K. Albulayhi, Q. Abu Al-Haija, S. A. Alsuhibany, A. A. Jillepalli, M. Ashrafuzzaman, and F. T. Sheldon, “IoT Intrusion Detection Using Machine Learning with a Novel High Performing Feature Selection Method,” Applied Sciences, vol. 12, no. 10, p. 5015, May 2022, doi: https://doi.org/10.3390/app12105015.

[10] S. Ali, Q. Li, and A. Yousafzai, “Blockchain and Federated Learning-based Intrusion Detection Approaches for Edge-enabled Industrial IoT Networks: A Survey,” Ad Hoc Networks, p. 103320, Oct. 2023, doi: https://doi.org/10.1016/j.adhoc.2023.103320.

[11] J. B. Awotunde, C. Chakraborty, and A. E. Adeniyi, “Intrusion Detection in Industrial Internet of Things Network-Based on Deep Learning Model with Rule-Based Feature Selection,” Wireless Communications and Mobile Computing, vol. 2021, p. e7154587, Sep. 2021, doi: https://doi.org/10.1155/2021/7154587.

[12] S. A. Bakhsh, M. A. Khan, F. Ahmed, M. S. Alshehri, H. Ali, and J. Ahmad, “Enhancing IoT network security through deep learning-powered Intrusion Detection System,” Internet of Things, vol. 24, no. 100936, p. 100936, Dec. 2023, doi: https://doi.org/10.1016/j.iot.2023.100936.

[13] S. Baniasadi, O. Rostami, D. Martín, and M. Kaveh, “A Novel Deep Supervised Learning-Based Approach for Intrusion Detection in IoT Systems,” Sensors, vol. 22, no. 12, p. 4459, Jun. 2022, doi: https://doi.org/10.3390/s22124459.

[14] Bassey Isong, Otshepeng Kgote, and Adnan Abu-Mahfouz, “Insights into Modern Intrusion Detection Strategies for Internet of Things Ecosystems,” Electronics, vol. 13, no. 12, pp. 2370–2370, Jun. 2024, doi: https://doi.org/10.3390/electronics13122370.

[15] Bilal Babayigit and M. Abubaker, “Towards a generalized hybrid deep learning model with optimized hyperparameters for malicious traffic detection in the Industrial Internet of Things,” Engineering applications of artificial intelligence, vol. 128, pp. 107515–107515, Feb. 2024, doi: https://doi.org/10.1016/j.engappai.2023.107515.

[16] J. Brownlee, “When to Use MLP, CNN, and RNN Neural Networks,” Machine Learning Mastery, Apr. 25, 2018. https://machinelearningmastery.com/when-to-use-mlp-cnn-and-rnn-neural-networks/

[17] N. Chockwanich and V. Visoottiviseth, “Intrusion Detection by Deep Learning with TensorFlow,” 2019 21st International Conference on Advanced Communication Technology (ICACT), Feb. 2019, doi: https://doi.org/10.23919/icact.2019.8701969.

[18] Danish Javeed, Muhammad Shahid Saeed, M. Adil, P. Kumar, and Alireza Jolfaei, “A federated learning-based zero trust intrusion detection system for Internet of Things,” Ad hoc networks, pp. 103540–103540, May 2024, doi: https://doi.org/10.1016/j.adhoc.2024.103540.

[19] A. Deshmukh and K. Ravulakollu, “An Efficient CNN-Based Intrusion Detection System for IoT: Use Case Towards Cybersecurity,” Technologies, vol. 12, no. 10, p. 203, Oct. 2024, doi: https://doi.org/10.3390/technologies12100203.

[20] E. Miguel et al., “A Hybrid CNN-LSTM Model for IIoT Edge Privacy-Aware Intrusion Detection,” Nov. 2022, doi: https://doi.org/10.1109/latincom56090.2022.10000468.

[21] E V N Jyothi, M Kranthi, S Sailaja, U Sesadri, S. N. Koka, and C. Shaker, “An Adaptive Intrusion Detection System in Industrial Internet of Things(IIoT) using Deep Learning,” Apr. 2024, doi: https://doi.org/10.1109/istems60181.2024.10560245.

[22] K. Elmazi, D. Elmazi, and J. Lerga, “A Survey on Fault Detection in Industrial IoT: A Machine Learning Approach with Emphasis on Federated Learning and Intrusion Detection Systems,” Jun. 2024, doi: https://doi.org/10.21203/rs.3.rs-4520887/v1.

[23] O. Elnakib, E. Shaaban, M. Mahmoud, and K. Emara, “EIDM: deep learning model for IoT intrusion detection systems,” The Journal of Supercomputing, Mar. 2023, doi: https://doi.org/10.1007/s11227-023-05197-0.

[24] “Enhancing Industrial IoT Security: Utilizing Blockchain- Assisted Deep Federated Learning for Collaborative Intrusion Detection - ProQuest,” Proquest.com, 2024. https://www.proquest.com/openview/582a990a86a3c86f3fb783de1761034e/1?pq-origsite=gscholar&cbl=4433095

[25] O. Friha, M. A. Ferrag, M. Benbouzid, T. Berghout, B. Kantarci, and K.-K. R. Choo, “2DF-IDS: Decentralized and differentially private federated learning-based intrusion detection system for industrial IoT,” Computers & Security, vol. 127, p. 103097, Apr. 2023, doi: https://doi.org/10.1016/j.cose.2023.103097.

[26] T. Gaber, J. B. Awotunde, S. O. Folorunso, S. A. Ajagbe, and E. Eldesouky, “Industrial Internet of Things Intrusion Detection Method Using Machine Learning and Optimization Techniques,” Wireless Communications and Mobile Computing, vol. 2023, p. e3939895, Apr. 2023, doi: https://doi.org/10.1155/2023/3939895.

[27] D. Hamouda, Mohamed Amine Ferrag, Nadjette Benhamida, Hamid Seridi, and Mohamed Chahine Ghanem, “Revolutionizing intrusion detection in industrial IoT with distributed learning and deep generative techniques,” Internet of Things, vol. 26, pp. 101149–101149, Jul. 2024, doi: https://doi.org/10.1016/j.iot.2024.101149.

[28] Hani Alshahrani, A. Khan, M. Rizwan, M. Saleh, A. Sulaiman, and Luige Vladareanu, “Intrusion Detection Framework for Industrial Internet of Things Using Software Defined Network,” vol. 15, no. 11, pp. 9001–9001, Jun. 2023, doi: https://doi.org/10.3390/su15119001.

[29] K. Hassini, S. Khalis, O. Habibi, M. Chemmakha, and M. Lazaar, “An end-to-end learning approach for enhancing intrusion detection in Industrial-Internet of Things,” Knowledge-Based Systems, vol. 294, p. 111785, Apr. 2024, doi: https://doi.org/10.1016/j.knosys.2024.111785.

[30] Himanshu Nandanwar and Rahul Katarya, “Deep learning enabled intrusion detection system for Industrial IOT environment,” Expert systems with applications, vol. 249, pp. 123808–123808, Sep. 2024, doi: https://doi.org/10.1016/j.eswa.2024.123808.

[31] Vanlalruata Hnamte and J. Hussain, “Network Intrusion Detection using Deep Convolution Neural Network,” May 2023, doi: https://doi.org/10.1109/incet57972.2023.10170202.

[32] Egor Ichetovkin and I. Kotenko, “Modeling Attacks on Machine Learning Components of Intrusion Detection Systems,” pp. 261–266, Mar. 2024, doi: https://doi.org/10.1109/smartindustrycon61328.2024.10515506.

[33] J. Alwina Beauty Angelin and C. Priyadharsini, “Deep Learning based Network based Intrusion Detection System in Industrial Internet of Things,” Jan. 2024, doi: https://doi.org/10.1109/idciot59759.2024.10467510.

[34] A. Arokiaraj Jovith, C. S. Ranganathan, S. Priya, R. Vijayakumar, R. Kohila, and S. Prakash, “Industrial IoT Sensor Networks and Cloud Analytics for Monitoring Equipment Insights and Operational Data,” Apr. 2024, doi: https://doi.org/10.1109/iccsp60870.2024.10543619.

[35] Joseph Bamidele Awotunde et al., “An Ensemble Tree-Based Model for Intrusion Detection in Industrial Internet of Things Networks,” Applied sciences, vol. 13, no. 4, pp. 2479–2479, Feb. 2023, doi: https://doi.org/10.3390/app13042479.

[36] A. Kaur, “Intrusion Detection Approach for Industrial Internet of Things Traffic using Deep Recurrent Reinforcement Learning Assisted Federated Learning,” IEEE Transactions on Artificial Intelligence, pp. 1–13, 2024, doi: https://doi.org/10.1109/tai.2024.3443787.

[37] A. Khacha, Rafika Saadouni, Y. Harbi, and Zibouda Aliouat, “Hybrid Deep Learning-based Intrusion Detection System for Industrial Internet of Things,” Nov. 2022, doi: https://doi.org/10.1109/isia55826.2022.9993487.

[38] N. Khan, K. Ahmad, A. A. Tamimi, M. M. Alani, A. Bermak, and I. Khalil, “Explainable AI-based Intrusion Detection System for Industry 5.0: An Overview of the Literature, associated Challenges, the existing Solutions, and Potential Research Directions,” arXiv.org, 2024. https://arxiv.org/abs/2408.03335

[39] H. Kim and K. Lee, “IIoT Malware Detection Using Edge Computing and Deep Learning for Cybersecurity in Smart Factories,” Applied Sciences, vol. 12, no. 15, p. 7679, Jul. 2022, doi: https://doi.org/10.3390/app12157679.

[40] B. R. Kikissagbe and M. Adda, “Machine Learning-Based Intrusion Detection Methods in IoT Systems: A Comprehensive Review,” Electronics, vol. 13, no. 18, p. 3601, Sep. 2024, doi: https://doi.org/10.3390/electronics13183601.

[41] H. Li, Gopi Krishnan Rajbahadur, and Cor-Paul Bezemer, “Studying the Impact of TensorFlow and PyTorch Bindings on Machine Learning Software Quality,” ACM Transactions on Software Engineering and Methodology, Jul. 2024, doi: https://doi.org/10.1145/3678168.

[42] X. Liu, S. Hu, and D. Yan, “A statistical quantitative analysis of the correlations between socio-demographic characteristics and household occupancy patterns in residential buildings in China,” Energy and Buildings, vol. 284, pp. 112842–112842, Apr. 2023, doi: https://doi.org/10.1016/j.enbuild.2023.112842.

[43] W. Li and Nazila Mohammadnezhad, “Improvement of intrusion detection system in industrial Internet of Things based on deep learning with fog computing capability,” Electronic Commerce Research, Dec. 2024, doi: https://doi.org/10.1007/s10660-024-09932-4.

[44] Y. Lu, S. Chai, Y. Suo, F. Yao, and C. Zhang, “Intrusion detection for Industrial Internet of Things based on deep learning,” Neurocomputing, vol. 564, pp. 126886–126886, Jan. 2024, doi: https://doi.org/10.1016/j.neucom.2023.126886.

[45] B. Madhu, M. Venu Gopala Chari, R. Vankdothu, A. K. Silivery, and V. Aerranagula, “Intrusion detection models for IOT networks via deep learning approaches,” Measurement: Sensors, vol. 25, p. 100641, Feb. 2023, doi: https://doi.org/10.1016/j.measen.2022.100641.

[46] Mangayarkarasi Ramaiah and Mohemmed Yousuf Rahamathulla, “Securing the Industrial IoT: A Novel Network Intrusion Detection Models,” May 2024, doi: https://doi.org/10.1109/aiiot58432.2024.10574728.

[47] D. Manivannan, “Recent endeavors in machine learning-powered intrusion detection systems for the Internet of Things,” Journal of Network and Computer Applications, vol. 229, p. 103925, Jun. 2024, doi: https://doi.org/10.1016/j.jnca.2024.103925.

[48] L. Mohammadpour, T. C. Ling, C. S. Liew, and A. Aryanfar, “A Survey of CNN-Based Network Intrusion Detection,” Applied Sciences, vol. 12, no. 16, p. 8162, Aug. 2022, doi: https://doi.org/10.3390/app12168162.

[49] O.-C. Novac et al., “Analysis of the Application Efficiency of TensorFlow and PyTorch in Convolutional Neural Network,” Sensors, vol. 22, no. 22, p. 8872, Nov. 2022, doi: https://doi.org/10.3390/s22228872.

[50] S. Nayak, N. Ahmed, and S. Misra, “Deep Learning-Based Reliable Routing Attack Detection Mechanism for Industrial Internet of Things,” Ad Hoc Networks, vol. 123, p. 102661, Dec. 2021, doi: https://doi.org/10.1016/j.adhoc.2021.102661.

[51] Qasem Abu Al-Haija and Ayat Droos, “A comprehensive survey on deep learning-based intrusion detection systems in Internet of Things (IoT),” Expert Systems, Sep. 2024, doi: https://doi.org/10.1111/exsy.13726.

[52] S. H. Rafique, A. Abdallah, N. S. Musa, and T. Murugan, “Machine Learning and Deep Learning Techniques for Internet of Things Network Anomaly Detection—Current Research Trends,” Sensors, vol. 24, no. 6, p. 1968, Jan. 2024, doi: https://doi.org/10.3390/s24061968.

[53] M. M. Rashid, S. U. Khan, F. Eusufzai, Md. A. Redwan, S. R. Sabuj, and M. Elsharief, “A Federated Learning-Based Approach for Improving Intrusion Detection in Industrial Internet of Things Networks,” Network, vol. 3, no. 1, pp. 158–179, Jan. 2023, doi: https://doi.org/10.3390/network3010008.

[54] Md. F. A. Sayeedi, J. F. Deepti, A. M. I. M. Osmani, T. Rahman, S. S. Islam, and Md. M. Islam, “A Comparative Analysis for Optimizing Machine Learning Model Deployment in IoT Devices,” Applied Sciences, vol. 14, no. 13, p. 5459, Jun. 2024, doi: https://doi.org/10.3390/app14135459.

[55] F. Sangoleye, J. Johnson, and E. Eleni Tsiropoulou, “Intrusion Detection in Industrial Control Systems Based on Deep Reinforcement Learning,” IEEE Access, vol. 12, pp. 151444–151459, 2024, doi: https://doi.org/10.1109/access.2024.3477415.

[56] S. J. Sam and S. Basil. Xavier, “Intrusion Detection System for Industrial IoT,” pp. 1–6, Jul. 2024, doi: https://doi.org/10.1109/icait61638.2024.10690722.

[57] J.-M. Shao, G.-Q. Zeng, K.-D. Lu, G.-G. Geng, and J. Weng, “Automated federated learning for intrusion detection of industrial control systems based on evolutionary neural architecture search,” Computers & Security, vol. 143, p. 103910, Aug. 2024, doi: https://doi.org/10.1016/j.cose.2024.103910.

[58] E. Sisinni, A. Saifullah, S. Han, U. Jennehag, and M. Gidlund, “Industrial Internet of Things: Challenges, Opportunities, and Directions,” IEEE Transactions on Industrial Informatics, vol. 14, no. 11, pp. 4724–4734, Nov. 2018, doi: https://doi.org/10.1109/tii.2018.2852491.

[59] S. Soliman, W. Oudah, and A. Aljuhani, “Deep learning-based intrusion detection approach for securing industrial Internet of Things,” Alexandria Engineering Journal, vol. 81, pp. 371–383, Oct. 2023, doi: https://doi.org/10.1016/j.aej.2023.09.023.

[60] I. A. Soomro, Hamood, S. J. Hussain, Z. Ashraf, M. M. Alnfiai, and N. N. Alotaibi, “Lightweight privacy-preserving federated deep intrusion detection for industrial cyber-physical system,” Journal of Communications and Networks, vol. 26, no. 6, pp. 632–649, Dec. 2024, doi: https://doi.org/10.23919/jcn.2024.000054.

[61] Y. Sun, C. Liu, Y. Weng, and Y. Liu, “Federated learning-based intrusion detection system for industrial Internet of Things: enhancing security and efficiency,” pp. 50–50, Jan. 2025, doi: https://doi.org/10.1117/12.3052237.

[62] K. Tsiknas, D. Taketzis, K. Demertzis, and C. Skianis, “Cyber Threats to Industrial IoT: A Survey on Attacks and Countermeasures,” IoT, vol. 2, no. 1, pp. 163–186, Mar. 2021, doi: https://doi.org/10.3390/iot2010009.

[63] S. Tharewal, M. W. Ashfaque, S. S. Banu, P. Uma, S. M. Hassen, and M. Shabaz, “Intrusion Detection System for Industrial Internet of Things Based on Deep Reinforcement Learning,” Wireless Communications and Mobile Computing, vol. 2022, pp. 1–8, Mar. 2022, doi: https://doi.org/10.1155/2022/9023719.

[64] D. Torre, Anitha Chennamaneni, J. Jo, G. Vyas, and B. Sabrsula, “Towards Enhancing Privacy-Preservation of a Federated Learning CNN Intrusion Detection System in IoT: Method and Empirical Study,” ACM Transactions on Software Engineering and Methodology, Sep. 2024, doi: https://doi.org/10.1145/3695998.

[65] Petros Toupas, Dimitra Chamou, K. M. Giannoutakis, Anastasios Drosou, and Dimitrios Tzovaras, “An Intrusion Detection System for Multi-class Classification Based on Deep Neural Networks,” Dec. 2019, doi: https://doi.org/10.1109/icmla.2019.00206.

[66] I. Ullah and Q. H. Mahmoud, “Design and Development of a Deep Learning-Based Model for Anomaly Detection in IoT Networks,” IEEE Access, vol. 9, pp. 103906–103926, 2021, doi: https://doi.org/10.1109/access.2021.3094024.

[67] J. Wang, C. Si, Z. Wang, and Q. Fu, “A New Industrial Intrusion Detection Method Based on CNN-BiLSTM,” Computers, materials & continua/Computers, materials & continua (Print), vol. 79, no. 3, pp. 4297–4318, Jan. 2024, doi: https://doi.org/10.32604/cmc.2024.050223.

[68] K. Yang, J. Wang, and M. Li, “An improved intrusion detection method for IIoT using attention mechanisms, BiGRU, and Inception-CNN,” Scientific Reports, vol. 14, no. 1, Aug. 2024, doi: https://doi.org/10.1038/s41598-024-70094-2.

[69] Tolulope Onasanya, “Enhancing Cyber Resilience with AI-Powered Cloud Threat Detection: A Multi-Layered Defense Approach,” Jun. 14, 2024. https://www.researchgate.net/publication/391111632_Enhancing_Cyber_Resilience_with_AI-Powered_Cloud_Threat_Detection_A_Multi-Layered_Defense_Approach

[70] Tolulope Onasanya, “Design Proposal of a Cloud-Based AI System for Real-Time Cyber Threat Detection and Autonomous Response,” Aug. 23, 2024. https://www.researchgate.net/publication/391050361_Design_Proposal_of_a_Cloud-Based_AI_System_for_Real-Time_Cyber_Threat_Detection_and_Autonomous_Response