

Original Article

Towards Robust Industrial IoT Security based on Artificial Intelligence Approach for Intrusion Detection Networks

Alpeshkumar Kathiriya

Independent Researcher

Received Date: 23 March 2025

Revised Date: 02 May 2025

Accepted Date: 15 June 2025

Abstract: The rise of automation in industry now makes protecting IoT networks very important. A flexible and effective network intrusion detection system (NIDS) helps reduce the number of cyber threats. In such ever-changing environments, traditional Intrusion Detection Systems (IDSs) failing to identify new or changing threats is a common occurrence. This research explores how to improve network security via accurate intrusion detection using AI approaches, particularly Machine Learning (ML) and Deep Learning (DL). The detection accuracy and efficiency were enhanced by using a Convolutional Neural Network (CNN) model that used modern preprocessing approaches, including SMOTE for data balance and PCA for feature selection. The results from using the CICIDS-2017 dataset showed that the suggested model scored an F1score of 99.88%, a recall of 99.51%, a precision of 98.16% and an accuracy of 98.81%. The model's capacity to distinguish between benign and malicious traffic with less false positives and missed dangers is shown by these measures. In comparison to regular ML models like AdaBoost, XGBoost, DecisionTree and RandomForest, the CNN approach delivered better performance, suggesting that it is quite robust and adaptable. This framework uses AI to offer a new and reliable way to defend IIoT networks by warning about and stopping cyberattacks in real time.

Keywords: Industrial Internet Of Things (IIoT), Cybersecurity, Intrusion Detection System (IDS), IIoT Security, Machine Learning (ML), Cicides2017 Dataset.

I. INTRODUCTION

The IIoT combines IoT with industrial systems so that now, processes in factories can be automated, controlled and watched live using connected sensors, actuators and devices. This merging of technology has transformed manufacturing, energy and transportation by making work more effective, efficient and straightforward. Because the IIoT ecosystem is growing, it becomes more exposed to cyberattacks [1][2]. IIoT threats include different forms of cyberattacks such as leaking data, gaining access without permission, tampering with programmed processes and stopping services[3] [4]. Such incidents can seriously damage systems, especially in locations such as nuclear power plants and automated lines where ensuring safety, stability and proper data is most urgent[5][6]. Many traditional approaches to security cannot cope with the diversity and constant change in IIoT and cause a greater demand for superior security options[7][8][9].

One of the core components of modern cybersecurity architecture in IIoT is the Intrusion Detection System (IDS)[10][11]. An IDS constantly monitors system activity and finds unusual signs that could be security problems. Yet, due to the massive amount and speed of data in IIoT and the intricate patterns of action, most traditional IDSs struggle with timely and accurate threat detection[12][13][14]. Realizing these boundaries, researchers are now trying to spot the patterns in IIoT communication that let them distinguish between safe and harmful activities[15][16][17]. It is important to spot minor deviations as soon as possible to avoid disasters. Effective methods for finding attacks should spot known threats as well as be ready for new and unknown risks[18][19]. By integrating ML and DL into IDSs, we can provide a promising solution for this purpose[20]. These methods allow machines to learn from a large amount of data from IIoT, respond to new types of cyberattacks and become more accurate without needing extensive human input[21]. Today, the use of ML and DL is best for noticing hidden problems and unusual situations in complex systems, so it is a perfect fit for protecting IIoT.

A. Motivation and Contribution of Paper

An essential security need is intrusion detection due to the growing vulnerability of Industrial IoT devices to cyberattacks. Traditional approaches find it difficult to identify complicated threats in changing environments. The need for an intelligent, precise, and scalable intrusion detection system that uses deep learning, more especially, a CNN model enhanced by efficient feature selection, data preprocessing, and class balancing techniques, to increase detection accuracy and safeguard vital infrastructure is what spurred this research. This research enhances IoT Security by integrating advanced preprocessing, optimized feature selection, and a DL-based model to improve Intrusion Detection.

- Leveraged the CICIDS2017 dataset, simulating realistic network traffic scenarios, to test and validate the effectiveness of the proposed intrusion detection method.



- Systematic handling of missing values, outlier removal, noise reduction, and conversion of categorical data using one-hot encoding to ensure data quality.
- Used SMOTE to address class imbalance, enhancing the model's ability to detect minority-class intrusions.
- Applied Principal Component Analysis to reduce feature space and highlight the most informative attributes, improving model efficiency and interpretability.
- Scaled all features to a common range to optimize neural network convergence and performance.
- Developed a CNN-based IDS tailored for Industrial IoT environments, achieving high accuracy and robustness.
- Accuracy, precision, recall, and the F1score derived from the confusion matrix should be used to assess the model's performance.

B. Justification and Novelty of Paper

The suggested CNN-based intrusion detection system is justified by its capacity to overcome the drawbacks of conventional IDS models, which often depend on preset rules and are unable to identify new or complex assaults in dynamic IIoT environments. The uniqueness of this method is located in its combination of deep learning's capacity to autonomously extract high-level features from raw traffic data with sophisticated data preparation methods like SMOTE for class balance and PCA for feature selection. Unlike conventional models, the proposed CNN approach achieves higher accuracy and better generalization by capturing both local and global network patterns. Validated on the CICIDS-2017 dataset, it outperforms traditional machine learning methods, offering a reliable and adaptive security solution. Its novelty lies in combining advanced preprocessing, dimensionality reduction, and DL for effective intrusion detection in industrial environments.

C. Structure of Paper

This is the outline for the remaining sections of the paper. Section II provides a background study on Intrusion Detection for Robust Industrial IoT Security. In Section III, the methodology and model implementation is detailed. In Section IV, the results, analysis, and discussion are compared. Section V presents the study's conclusion and plans for further research.

II. LITERATURE REVIEW

This section provides an overview of previous work on improving network security via network intrusion detection (NID) utilising AI approaches.

Zhou et al., (2025) come up with CBCTL-IDS, an innovative transfer learning-based network intrusion detection system. A Confidence Averaging Mechanism, a BlackKite Algorithm (BKA), and CNN are the three mainstays of this approach. The CBCTL-IDS approach outperforms current mainstream approaches by achieving detection accuracy rates of over 99% on 3 IoT intrusion detection datasets: ToN-IoT, Edge-IIoTset, and WSN-DS, according to experimental findings. Strong technological assistance for IoT system security is offered by this method[22].

Na, Haldorai and Naik, (2025) analyses are performed on satellite data to identify any significant changes which the model uses to point out potential threats. Furthermore, the information's temporal dependencies may be stopped over time. After enhancing the accuracy of the universal learning strategy, this network gathers data from many sources. The suggested method has a much better detection ratio, and experimental findings demonstrate that it can successfully identify assaults on the bus in real-time. The investigation indicates that the hybrid model achieves 99.94% accuracy and 99.86% precision on the HCRL SA dataset. Recall and area of the curve are also proven to perform better when the suggested strategy is used. It is 99.10 percent and 99.21 percent, respectively[23].

Alrayes et al., (2024) proposed method builds a system that can recognise and stop infiltration attempts in real-time by using the unsupervised learning and feature extraction capabilities of DAEs. The NSL-KDD and CICIDS 2017 datasets are also used in the study's assessment. Using the CICIDS 2017 dataset, DAE integration produces an unparalleled accuracy of 99.991%, while using the NSL-KDDdataset, it produces an accuracy of 99.4%. The data from the CICIDS 2017 dataset highlights that the model had an F1score of 0.998, a precision of 0.995 and an accuracy of 1.0. The analysis of the NSL-KDD dataset recorded an F1score of 0.989, recall of 0.991, accuracy of 0.994 and precision of 0.984[24].

Ramaiah and Rahamathulla, (2024) analyzes the advantages of LSTM and other ML techniques for preventing planned attacks on Industrial IoT (IIoT) networks. Having an appropriate dataset is necessary for effective NIDS design. The proposed technique surpasses the present state-of-the-art NIDS, according to the experimental findings. The cyber-attack detection accuracy for the ERT-based IIoT-NIDS was 99.93%, while for the LSTM-based IIoT-NIDS it was 99.85% [25].

Zukaib et al., (2024) strategy uses both signature detection and anomaly detection, while also including important privacy features for IoMT data. The findings show that the detection methods used achieve outstanding accuracy rates of 99.47%, 99.98%, and 99.99% when using anomaly detection, and 99.93% and 99.99% when using signature detection, respectively. Misclassification occurs infrequently, at rates of 0.0042%, 0.0006% and 0.00004%. Meta-IDS prove to be

effective and reliable, based on comparisons with the newest E-GraphSAGE model and measured by accuracy, precision, recall, F1-score, time complexity and misclassification rate[26].

Table 1 presents the relevant background information on IntrusionDetection for Robust Industrial IoT Security, including its dataset, models, performance, and contribution.

Table 1 : Overview of Recent Studies on Intrusion Detection Networks on Artificial Intelligence

Author	Proposed Work	Dataset	Key Findings	Challenges/Gaps
Zhou et al., (2025)	CBCTL-IDS based on transfer learning integrating CNN, Black Kite Algorithm, and Confidence Averaging	ToN-IoT, Edge-IIoTset, WSN-DS	Achieved detection accuracy > 99%, outperforming mainstream methods; robust and stable detection via ensemble learning	Need for evaluation on more diverse IoT datasets; complexity of integrating multiple components
Na, Haldorai and Naik (2025)	Hybrid model extracting spatial and temporal features for real-time bus attack detection	HCRL SA	High accuracy and precision (99.94%, 99.86%), improved recall and AUC (99.10%, 99.21%)	Real-time scalability and computational efficiency in resource-constrained environments
Alrayes et al., (2024)	Unsupervised Denoising Autoencoders (DAEs) for real-time intrusion detection	CICIDS2017, NSL-KDD	Exceptional accuracy up to 99.991% on CICIDS2017, with high precision and F1-score; also strong performance on NSL-KDD	Generalization to new attack types; computational cost of DAE integration
Ramaiah and Rahamathulla (2024)	DL and ML models (ERT, LSTM) for IIoT cyber-attack detection	EdgeIIoT-2021	Achieved high detection accuracy: ERT (99.93%), LSTM (99.85%), outperforming existing NIDS	Handling concept drift in dynamic IIoT environments
Zukaib et al., (2024)	Meta-IDS integrating signature-based and anomaly-based detection with privacy-preserving techniques	WUSTL-EHMS-2020, IoTID20, WUSTL-IIOT-2021	Remarkable accuracy (~99.5% to 99.99%) with low misclassification; competitive with E-GraphSAGE; robust for IoMT network security	Complexity in privacy preservation and scalability in large IoMT networks
Chen et al., (2023)	AI-based autonomous abnormal network traffic detection and response mechanism (AI BOX)	Two public datasets	Achieved 99.9% prediction accuracy; capable of interrupting/modifying anomalous traffic in real-time; supports heterogeneous networks	Integration challenges in heterogeneous IT and OT networks
Rizvi et al., (2023)	AI models trained and deployed on resource-constrained devices for forensic readiness	IoT-23	Classification accuracy over 99% on constrained devices; effective real-time detection with minimal overhead	Limited computational resources; need for continuous model updates
Bagaa et al., (2020)	ML-based security framework for IoT with anomaly-based IDS using one-class SVM	Real Smart Building Scenario	Achieved anomaly detection accuracy of 99.71%; efficient and low-cost detection	Scalability to large-scale IoT deployments; handling evolving threats

Chen et al., (2023) offers a method for identifying intrusions that is based on AI. Using AI-based algorithms, the system examines captured packets to identify malicious network activity or suspicious network traffic. In order to identify outliers, AI algorithms are used. AI BOX environment settings, packet tracking capabilities, and ML models have all been built by IT. It achieved a prediction accuracy of 99.9 percent after training the model with two publicly available datasets. The AI BOX gadget was able to evaluate the performance of several AI models[27].

Rizvi et al., (2023) analyses a method that puts AI models through their paces on devices with limited resources, safeguarding networks and marking important traffic for further examination. With little overhead, our technique detects and records potential hostile attacks in real-time, making it perfect for constrained contexts. The experiments were conducted using the IoT-23 dataset. The findings showed that on a sample device with limited resources, all of the algorithms achieved classification accuracy levels over 99% [28].

Bagaa et al., (2020) introduces a unique security architecture that uses ML to automatically handle the growing number of security concerns in the IoT sector. The proposed approach is effective, according to the experimental data. Specifically, the data mining technique to attack distribution is very effective in identifying assaults with minimal cost and great performance. In a real-world smart building situation, the experiment was evaluated using one-class SVM in relation to anomaly-based IDS for the Internet of Things. Anomaly detection accuracy reached 99.71%. In order to further research towards unanswered problems and determine whether existing solutions are feasible, a feasibility study is carried out[29].

III. RESEARCH METHODOLOGY

To secure Industrial IoT networks, an AI-driven IDS is introduced using the CICIDS-2017dataset. At the start, we look at network activity closely, using heatmaps to find relationships between features and boxplots to spot unusual values. After that, the data is cleaned by fixing missing entries, removing problems caused by noise, excluding outliers and converting categorical data with one-hot encoding. To tackle class imbalance, the SMOTE technique generates synthetic examples of minority classes, promoting balanced learning. PCA is employed for feature selection, reducing complexity while retaining key patterns. The dataset undergoes normalization via Min-Max scaling and is split into training and testing subsets. At the core of the detection system is a CNN, designed to recognize and classify network threats by learning intricate data patterns. Performance is assessed through metrics such as accuracy, precision, recal, and F1score, using a ConfusionMatrix to capture the full picture of classification outcomes. The scalable and efficient solution provided by this organised, AI-powered architecture protects IIoT environments from changing cyberthreats. The implementation steps illustrate in figure 1.

The following contains thorough descriptions of every step in a flowchart.

A. Data Collection

The data collection used to train the IDS is the CICIDS-2017dataset. Using protocols such as HTTP, HTTPS, FTP, SSH, and email, the dataset simulates the behaviours of twenty-five individuals who unknowingly create harmless traffic. Includes common 2016 attack scenarios including brute force, denial of service, distributed denial of service, infiltration, Heartbleed, botnet, and port scan methods. The correlated features of dataset are illustrate in figure 1.

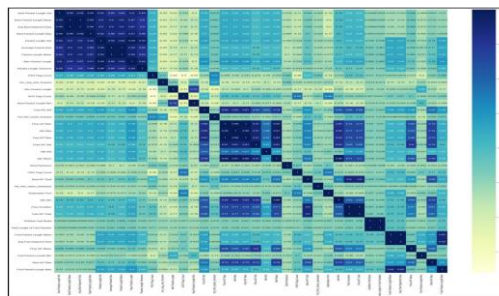


Figure 1 : Heatmap of the Features

An example of a heatmap representation of a correlation matrix with values between -1 and 1 is shown in Figure 2. The correlation coefficient between two variables is shown in each cell of the matrix. A greater positive correlation is indicated by a brighter blue shade, whereas a weaker negative correlation is indicated by a lighter yellowish hue. Each variable is perfectly correlated with itself (value=1) along the diagonal line that runs from top-left to bottom-right. Groups of strongly linked variables are suggested by the clustered black blocks, which may indicate common characteristics or patterns within the data. The right-hand colour bar serves as a benchmark for understanding the significance of the colours in regard to the strength of the association.

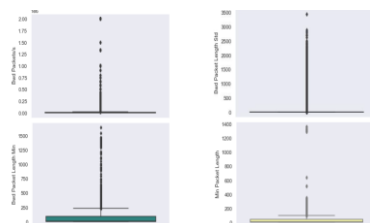


Figure 2 : Boxplots of Outliers

Figure 2 displays four box plots that illustrate the distribution and presence of outliers in four different variables: "Band Probabilities," "Band Pocket Length Std," "Band Pocket Length Min," and "Min Pocket Length." Each plot shows a large number of extreme values (outliers), which are indicated by individual dots above the whiskers. The bulk of the data for each variable is concentrated near the lower end of the range, suggesting highly skewed distributions. The whiskers and boxes are very compressed, reflecting that most data points lie close to the lower values, while a few extremely high values stretch the scale upward. This visualization highlights the need for potential normalization or transformation steps during data preprocessing.

B. Data Pre-Processing

Data preparation is just transforming unprocessed data into a format that is easier to understand [30]. There are instances of noisy, inconsistent, duplicate, or incomplete real-world data. Data preparation is a series of steps that transform raw data into a usable format. Following is a rundown of the pre-processing steps:

- Handle missing value: The knowledge that the mean value of a numeric column is used to fill in missing values, thereby maintaining the central tendency, is essential.
- Remove Outliers: An outlier is a data point that is unusually distributed in relation to other points. In essence, it describes information that is different from the other values in the collection. Outliers may cause true findings to be misrepresented or a major discovery to be overlooked, which is an issue for many statistical studies.
- Remove noise: Taking the median and interquartile range into account helps provide better results when trying to remove noise caused by outliers.

C. One-Hot Encoding for Data Labeling

Text strings and categorical variables must be converted into numerical ones in order for an MLmodel to calculate their connection and provide accurate predictions, because the majority of MLmodels only comprehend numbers [31]. To solve this, apply one-hot encoding techniques. Binary vectors in the one-hot encoding provide categorical feature variables their numerical values (0 or 1). Compared to traditional date systems, this facilitates the manipulation and storage of dated data by computer systems.

D. Balancing with SMOTE

The SMOTE is a method for fixing imbalanced datasets that entails making fictitious samples of the minority group. Relying on interpolation rather than duplication to produce fresh samples, it guarantees a more consistent distribution of classes. Because of this, ML models are better able to deal with imbalanced categorisation issues.

E. PCA for Feature Selection

A critical part of ML is feature selection, which is picking out the best features from a dataset. To extract useful characteristics from high-dimensional data, PCA often is the method of choice. You may use it to make complicated data sets easier to understand, find trends in the data, and develop better machine learning models. The following feature importance score graph are provide in below:

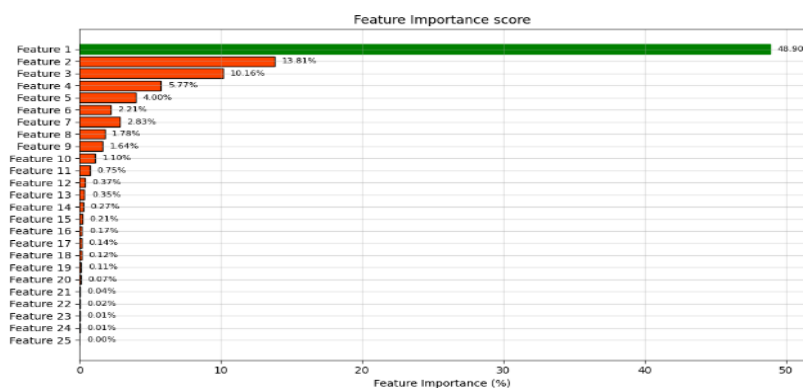


Figure 3 : Feature Importance Score

Figure 3 shows the relative value of 25 distinct traits, arranged in decreasing order. "Feature 1" is by far the most important, accounting for 48.90% of the total importance, significantly more than any other feature. "Feature 2" and "Feature 3" follow with 13.81% and 10.16% respectively, while the importance rapidly diminishes for subsequent features. Many features, particularly from "Feature 19" onwards, contribute less than 0.1% to the overall importance, with "Feature 25" showing 0.00% importance. The chart clearly highlights a few dominant features and a long tail of features with minimal individual impact.

F. Max-Min Normalization

Data normalisation using features scaled to a specific range, usually [0, 1], is accomplished with the Min-Max Scaler. To carry out the transformation, the formula in equation (1) is used:

$$X_{normalized} = \frac{X - X_{min}}{X_{max} - X_{min}}$$

where the data feature's current value is represented by X and X_{min} is the data feature's lowest value and X_{max} is its highest value.

G. Data Splitting

The dataset is partitioned into two halves, with 70% and 30% of the total size allocated for training and testing, respectively.

H. Proposed Convolutional Neural Network (CNN)

Image processing and recognition are common applications of CNNs, a kind of DL method that use ANN to identify patterns in pictures [32]. There are convolutional, pooling and fully connected layers in a CNN. By using a pooling layer, the system removes a few of the input parameters and the convolutional layer changes images into numbers. The CNN works better than others when it comes to image processing and recognition. Fully connected, pooling and convolutional layers are examples of what make up a neural network[33]. Like how a brain organizes visual information, CNNs are made to detect spatial and hierarchical patterns in pictures which makes them perfect for image identification. The output size of a convolutional layer can be found by using variable t. JavaScript produces 5 digits of output here. Next, you deal with what length the output should be,

$$\text{Output size} = nx = 2P - nhS + 1,$$

where both the input signal length (nx) and the filter length (nh) are considered. Many computer vision, signal processing and image processing applications use the convolution operation (Conv_Op). A weighted third signal results from multiplying two signals or functions to represent the effect of the first signal on the second. CNNs play a big role in feature extraction from photographs in computer vision tasks. In mathematics, the convolution operation is defined as (3):

$$(f * g)[n] = \sum_{m=-\infty}^{\infty} f[m]g[n - m]$$

While n represents the position or time index of the output signal, f and g might be continuous or discrete functions. A representation of the convolution process is the symbol *. It is possible to rewrite the above equation as (4, 5) when working with discrete input signals:

$$(f * g)[n] = \sum_{m=-\infty}^{\infty} f[m]g[n - m]\Delta m$$

$$(f * m)(t) = \int_{-\infty}^{\infty} f(\tau)g(t - \tau)d\tau$$

where t is the time index of the output signal.

I. Evaluation Metrics

This is the final step of the prediction model. A performance metric for ML classification issues, where the output may consist of two or more classes, is the confusion matrix. This confusion matrix is shown graphically below as a table containing four separate sets of predicted and actual values. TN, FP, TP, and FN are some examples of these pairings. The following instances of the confusion matrix are

- TP (True Positive): for describing the correctly identified positive tuples by the classifier.
- FP (False Positive): Alludes to the positive tuples that the classifier mistakenly labelled.
- FN (False Negative): is used to characterise the negative tuples that the classifier incorrectly labelled.
- TN (True Negative): utilised to characterise the correctly labelled negative tuples by the classifier.

Accuracy: Accuracy is defined as the proportion of valid predictions relative to the total number of predictions in the test dataset. It is given as (6)-

$$Accuracy = \frac{TP + TN}{TP + Fp + TN + FN}$$

Precision: The percentage of positive classes that are accurately predicted is called the precision of a class prediction. The sum of all positive observations anticipated and the number of correct predictions is divided by this sum to get it. It is expressed as (7)-

$$Precision = \frac{TP}{TP + FP}$$

Recall: The recall is the percentage of all correct positive predictions among all the correct positive samples. In mathematical form, it is given as (8)-

$$Recall = \frac{TP}{TP + FN}$$

F1 score: A prediction model's F1-score is found by averaging recall and precision using harmonic mean. Mathematically, it is given as (9)-

$$F1 - score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

These metrics together give information about the model's accuracy and efficacy in predicting the target variable.

IV. RESULTS AND DISCUSSION

An experimental result obtained from the proposed Intrusion Detection is analyzed in this section. This experiment uses new equipment, including an 11th Generation Intel® Core™ i9-11900KF with 16 cores operating at 3.50GHz. With a strong 62.5GB RAM, this CPU leads to effective data management and helps process data simultaneously. We used the six evaluation metrics, namely, Accuracy, F-measure, Recall, and Precision for model evaluation as shown in Table 2. The proposed model was able to classify data with an accuracy of 98.8.1% which is truly remarkable for both benign and malicious data. The model is very precise at 98.16% which helps minimize unintended false positives and guarantees the majority of the identified intrusions are genuine threats. Its recall of 99.51% reflects a high capability to detect nearly all actual intrusion attempts, reducing the chances of missed threats. The CNN model is an excellent and trustworthy option for NID tasks, as shown by the F1score of 99.88%, which implies a balanced performance between recall and precision.

Table 2 : Experiments Results of Proposed Model on CICIDS-2017 Dataset for Intrusion Detection

Performance matrix	Convolutional Neural Network (CNNs)
Accuracy	98.81
Precision	98.16
Recall	99.51
F1-score	99.88

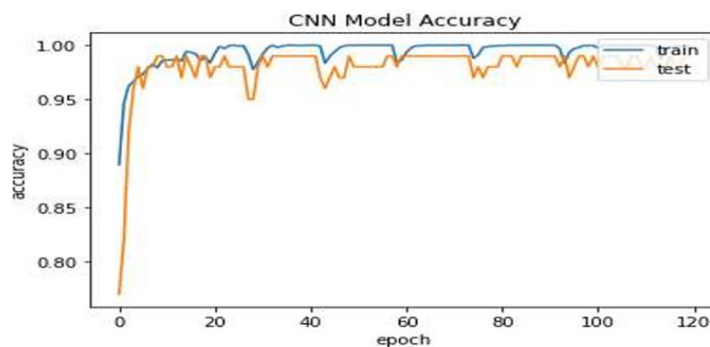


Figure 4 : Training and Testing Accuracy for the CNN Model

Figure 4 displays the recommended CNN's training and testing accuracy throughout 120 epochs. As the x-axis indicates a total number of epochs, the y-axis displays the accuracy values, which may be anywhere from 0.80 to 1.00. Visual inspection of the plot reveals that the model reaches a high level of accuracy in less than a decade, with testing and training accuracy levels above 95%. The blue line represents the training accuracy, which shows that the model is learning well without overfitting. It grows and stabilises at roughly 99%. Consistently high, the test accuracy (orange line) follows the training accuracy closely, with little fluctuations during the epochs. These findings point to the CNN model's strong performance during training and its good generalisability to new data.

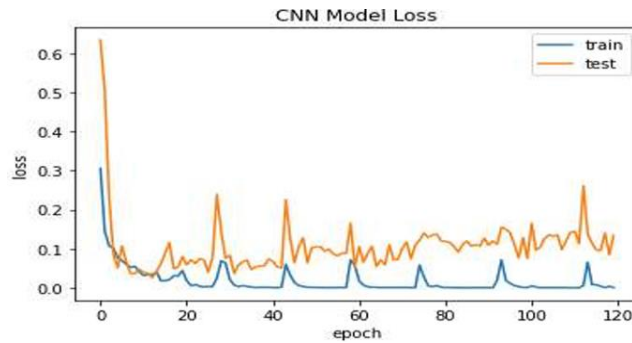


Figure 5 : Training and Testing Loss for the CNN Model

A patterns of the CNN's training and testing losses throughout 120 epochs are displayed in Figure 5. The y-axis shows loss values (from 0 to 0.5), while the x-axis displays the total number of epochs. At the outset, the blue line represents training losses and the orange line testing losses are both quite high, with the test loss peaking at around 0.5. A sharp drop in loss within the first ten epochs, however, suggests the model is learning quickly and performing well. As training progresses, both curves continue to decrease and stabilize, with occasional fluctuations in the test loss. These minor spikes, particularly after epoch 80, may be due to model sensitivity to certain test samples or temporary overfitting, but overall, the loss remains low, mostly below 0.1. This consistent reduction and stabilization of loss values suggest that the CNN model is learning effectively and maintaining strong generalization capabilities throughout the training process.

A. Comparison with Discussion

In this section, examine and compare our method with others' methods. The comparisons are based on the models' performance between the base and proposed models and are provided in Table III. In this comparison, AdaBoost model achieved an accuracy of 81.47%, a precision of 81.69%, a high recall of 95.76%, and an F1score of 88.17%, indicating strong detection capability but lower overall precision. XGBoost (XGB) performed better, with an accuracy of 94%, a precision of 96%, a recall of 92%, and an F1score of 94%, showing balanced and robust performance. The Decision Tree (DT) model achieved an accuracy of 95.40%, precision of 96.50%, recall of 93.30%, and F1score of 94.87%, slightly outperforming XGB. Random Forest (RF) showed high accuracy at 96%, but with slightly lower precision (89%), recall (88%), and F1score (88%) compared to DT. The proposed CNN model, representing the DL approach, delivered the best results overall, with the highest accuracy of 98.81%, precision of 98.16%, recall of 99.51%, and an F1score of 99.88%, highlighting its superior capability in accurately detecting and classifying intrusions in the dataset.

Table 3 : Performance Comparison of Machine Learning and Deep Learning Models for Intrusion Detection using the CICIDS-2017 Dataset

Models	Accuracy	Precision	Recall	F1-Score
AdaBoost [34]	81.47	81.69	95.76	88.17
XGB[35]	94	96	92	94
DT[36]	95.40	96.50	93.30	94.87
RF[37]	96	89	88	88
CNN	98.81	98.16	99.51	99.88

The proposed system has several important advantages for the security of Industrial IoT. It can learn from data by itself, helping it to adapt when new threats emerge, without having to build additional features. It identifies most threats with very little chance of a false alarm because it is precise. Using SMOTE and PCA, the learning becomes fair and efficient so the model can be applied to large amounts of data at any time. In general, it protects industrial networks with reliability, intelligence and efficiency.

V. CONCLUSION AND FUTURE STUDY

Industrial Control Systems (ICSs) are more susceptible to cyberattacks, which might have catastrophic results, due to their integration of communication networks and the IoT. The flexibility and efficacy of traditional Intrusion Detection Systems (IDSs) are severely limited in dynamic industrial situations due to their reliance on predetermined models and training on particular intrusions. These systems were primarily built to assist IT systems. To solve these issues, this paper introduces a CNN-based IDS that can recognize detailed patterns in the network using deep learning, requiring no separate features from network data. Training on the CICIDS-2017 dataset gave the model impressive outcomes with an accuracy of 98.81%, a precision of 98.16%, a recall of 99.51% and an F1score of 99.88%. These numbers outperform traditional ML models such as AdaBoost, XGBoost, DecisionTree and Random Forest. While it works well, machine learning still faces a few

problems, for example, the high costs of calculations, imbalances in data and not being able to respond well to new attacks. Future studies should focus on making intrusion detection models quicker, lighter and easy to adapt, while also updating the training data with recent cyberattack incidents. The enhancements will support the reliability, strength and growth of IDS solutions for the changing IIoT environment.

VI. REFERENCES

- [1] S. Singamsetty, "FUZZY-OPTIMIZED LIGHTWEIGHT CYBER-ATTACK DETECTION FOR SECURE EDGE-BASED IOT NETWORKS," *J. Crit. Rev.*, vol. 6, no. 7, pp. 1028-1033, 2019.
- [2] V. Thangaraju, "Security Considerations in Multi-Cloud Environments with Seamless Integration: A Review of Best Practices and Emerging Threats," *Trans. Eng. Comput. Sci.*, vol. 12, no. 2, p. 6, 2024.
- [3] U. Tariq, I. Ahmed, A. K. Bashir, and K. Shaukat, "A Critical Cybersecurity Analysis and Future Research Directions for the Internet of Things: A Comprehensive Review," 2023. doi: 10.3390/s23084117.
- [4] Z. A. El Houda, A. S. Hafid, and L. Khoukhi, "A Novel Machine Learning Framework for Advanced Attack Detection using SDN," in *Proceedings - IEEE Global Communications Conference, GLOBECOM*, 2021. doi: 10.1109/GLOBECOM46510.2021.9685643.
- [5] V. Prajapati, "Enhancing Threat Intelligence and Cyber Defense through Big Data Analytics: A Review Study," *J. Glob. Res. Math. Arch.*, vol. 12, no. 4, 2025.
- [6] A. Das Pushpalika Chatterjee, "AI-Powered Anomaly Detection for Real-Time Performance Monitoring in Cloud Systems," *Int. J. Sci. Res. Sci. Technol.*, vol. 11, no. 6, p. 10.32628, 2024.
- [7] N. G. Abhinav Balasubramanian, "Building secure cybersecurity infrastructure: integrating ai and hardware for real-time threat analysis," *Int. J. Core Eng. Manag.*, vol. 6, no. 7, pp. 263-270, 2020.
- [8] S. Chatterjee, "Risk Management in Advanced Persistent Threats (APTs) for Critical Infrastructure in the Utility Industry," *Int. J. Multidiscip. Res.*, vol. 3, no. 4, pp. 1-10, 2021.
- [9] M. I. Khan, A. Arif, and A. R. A. Khan, "AI-Driven Threat Detection: A Brief Overview of AI Techniques in Cybersecurity," *BIN Bull. Informatics*, vol. 2, no. 2, pp. 248-261, 2024.
- [10] P. Barnard, N. Marchetti, and L. A. DaSilva, "Robust Network Intrusion Detection Through Explainable Artificial Intelligence (XAI)," *IEEE Netw. Lett.*, 2022, doi: 10.1109/lnet.2022.3186589.
- [11] A. kumar Polinati, "AI-Powered Anomaly Detection in Cybersecurity: Leveraging Deep Learning for Intrusion Prevention," *Int. J. Commun. Networks Inf. Secur.*, vol. 17, no. 3, p. 13, 2025.
- [12] S. K. R. Mallidi and R. R. Ramisetty, "Advancements in training and deployment strategies for AI-based intrusion detection systems in IoT: a systematic literature review," vol. 5, no. 1. Springer International Publishing, 2025. doi: 10.1007/s43926-025-00099-4.
- [13] A. Amouri, V. T. Alaparthi, and S. D. Morgera, "A machine learning based intrusion detection system for mobile internet of things," *Sensors (Switzerland)*, 2020, doi: 10.3390/s20020461.
- [14] N. Prajapati, "Federated Learning for Privacy-Preserving Cybersecurity : A Review on Secure Threat Detection," pp. 520-528, 2025, doi: 10.48175/IJARSCT-25168.
- [15] G. Krishnamoorthy and S. M. K. Sistla, "Exploring Machine Learning Intrusion Detection: Addressing Security and Privacy Challenges in IoT - A Comprehensive Review," *J. Knowl. Learn. Sci. Technol.* ISSN 2959-6386, 2023, doi: 10.60087/jklst.vol2.n2.p125.
- [16] V. Thangaraju, "Enhancing Web Application Performance and Security Using AI-Driven Anomaly Detection and Optimization Techniques," *Int. Res. J. Innov. Eng. Technol.*, vol. 9, no. 3, p. 8, 2025.
- [17] N. Patel, "AI-Powered Intrusion Detection And Prevention Systems in 5G Networks," *Int. Conf. Commun. Electron. Syst. - ICCES-2024*, 2024.
- [18] K. He, D. D. Kim, and M. R. Asghar, "Adversarial Machine Learning for Network Intrusion Detection Systems: A Comprehensive Survey," *IEEE Commun. Surv. Tutor.* 2023, doi: 10.1109/COMST.2022.3233793.
- [19] P. M. Rajendra Prasad Sola, Nihar Malali, *Cloud Database Security: Integrating Deep Learning and Machine Learning for Threat Detection and Prevention*: o. 2025.
- [20] V. Pillai, "Anomaly Detection for Innovators: Transforming Data into Breakthroughs," 2022
- [21] R. Q. Majumder, "Machine Learning for Predictive Analytics: Trends and Future Directions," *Int. J. Innov. Sci. Res. Technol.*, vol. 10, no. 04, pp. 3557-3564, 2025.
- [22] H. Zhou, H. Zou, P. Zhou, Y. Shen, D. Li, and W. Li, "CBCTL-IDS: A Transfer Learning-Based Intrusion Detection System Optimized With the Black Kite Algorithm for IoT-Enabled Smart Agriculture," *IEEE Access*, vol. 13, pp. 46601-46615, 2025, doi: 10.1109/ACCESS.2025.3550800.
- [23] I.-S. Na, A. Haldorai, and N. Naik, "Federal Deep Learning Approach of Intrusion Detection System for In-Vehicle Communication Network Security," *IEEE Access*, vol. 13, pp. 2215-2228, 2025, doi: 10.1109/ACCESS.2024.3521661.
- [24] F. S. Alrayes, M. Zakariah, S. U. Amin, Z. Iqbal Khan, and M. Helal, "Intrusion Detection in IoT Systems Using Denoising Autoencoder," *IEEE Access*, vol. 12, pp. 122401-122425, 2024, doi: 10.1109/ACCESS.2024.3451726.
- [25] M. Ramaiah and M. Y. Rahamathulla, "Securing the Industrial IoT: A Novel Network Intrusion Detection Models," in *2024 3rd International Conference on Artificial Intelligence For Internet of Things (AIIoT)*, 2024, pp. 1-6. doi: 10.1109/AIIoT58432.2024.10574728.
- [26] U. Zukaib, X. Cui, C. Zheng, M. Hassan, and Z. Shen, "Meta-IDS: Meta-Learning-Based Smart Intrusion Detection System for Internet of Medical Things (IoMT) Network," *IEEE Internet Things J.*, vol. 11, no. 13, pp. 23080-23095, 2024, doi: 10.1109/JIOT.2024.3387294.

- [27] J. L. Chen et al., "AI BOX: Artificial intelligence-based autonomous abnormal network traffic response mechanism," in International Conference on Information Networking, 2023. doi: 10.1109/ICOIN56518.2023.10048979.
- [28] S. Rizvi, M. Scanlon, J. McGibney, and J. Sheppard, "An Evaluation of AI-Based Network Intrusion Detection in Resource-Constrained Environments," in 2023 IEEE 14th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference, UEMCON 2023, 2023. doi: 10.1109/UEMCON59035.2023.10315971.
- [29] M. Bagaa, T. Taleb, J. B. Bernabe, and A. Skarmeta, "A Machine Learning Security Framework for Iot Systems," IEEE Access, 2020, doi: 10.1109/ACCESS.2020.2996214.
- [30] N. Malali, "AI-Powered Data Preprocessing and Transformation Platform for Autonomous Data Cleaning, Advanced Fea," 202521035175, 2025
- [31] T. Al-Shehari and R. A. Alsowail, "An Insider Data Leakage Detection Using One-Hot Encoding, Synthetic Minority Oversampling and Machine Learning Techniques," Entropy, vol. 23, no. 10, 2021, doi: 10.3390/e23101258.
- [32] Y. H. Rajarshi Tarafdar, "Finding majority for integer elements," J. Comput. Sci. Coll., vol. 33, no. 5, pp. 187–191, 2018.
- [33] Sailaja Ayyalasomayajula, "A Mathematical Real Analysis on 2D Connection Spaces for Network Cyber Threats: A SEIAR-Neural Network Approach," Commun. Appl. Nonlinear Anal., vol. 31, no. 8s, pp. 179–198, Sep. 2024, doi: 10.52783/cana.v31.1474.
- [34] S. Soumik, "A comparative analysis of Network Intrusion Detection (NID) using Artificial Intelligence techniques for increase network security," 2024.
- [35] S. B. A. Maricar, A. Anoop, B. E. Samuel, A. Appukuttan, and K. H. Alsinjlawi, "An Improved Explainable Artificial Intelligence for Intrusion Detection System," Int. J. Intell. Syst. Appl. Eng., 2024.
- [36] K. Hasan, V. Guduru, S. Zein-Sabatto, D. Chimba, and I. Ahmed, "Towards Robust AI Model for Cyber-Physical Intelligent Transportation Systems by Hash-Based Ensemble Learning," in Proceedings - 2023 IEEE Latin-American Conference on Communications, LATINCOM 2023, 2023. doi: 10.1109/LATINCOM59467.2023.10361868.
- [37] S. Patil et al., "Explainable Artificial Intelligence for Intrusion Detection System," Electron., 2022, doi: 10.3390/electronics11193079.