International Journal of Current Engineering and Technology
E-ISSN 2277 – 4106, P-ISSN 2347 – 5161

Available at http://inpressco.com/category/ijcet

Research Article

Automated Threat Detection and Risk Mitigation for ICS (Industrial Control Systems) Employing Deep Learning in Cybersecurity Defence

Ruchi Patel*

Independent Researcher

Received 01 Dec 2023, Accepted 24 Dec 2023, Available online 26 Dec 2023, **Vol.13, No.6 (Nov/Dec 2023)**

Abstract

Industrial Control Systems (ICS) prove more susceptible to cyber threats which makes it necessary to create effective threat detection systems. The improvements in cybersecurity fields do not deliver sufficient scalability with real-time threat detection functionality. The paper designs an AI-based framework for ICS cybersecurity defense which applies deep learning to automate threat discovery along with risk reduction procedures. This research explores machine learning While deep learning (DL) models detect cyber threats using CICIDS-2017 dataset information. The testing phase included a CNN primary classification model against KNN traditional models and Naïve Bayes (NB) traditional models together with Support Vector Machine (SVM) traditional models. The CNN model exhibited the best performance by reaching 99.58% accuracy and precision as well as recall and F1-score resulting in its well-documented superiority in detecting cyber threats. The model achieved verification of its performance by examining accuracy curves and loss diagrams together with confusion matrix results. Deep learning has proven its effectiveness in industrial control system security by delivering sustainable real-time invasion detection capabilities along with risk management solutions.

Keywords: *Cybersecurity, Cyber Threats, Industrial Control Systems (ICS), Threat Detection, Risk Mitigation, Deep Learning (DL), CICIDS-2017.*

Introduction

The internet expansion alongside connected devices between systems created a transformation of industries, which delivers remarkable automation capabilities as well as improved efficiency. Digital transformation brought extensive cybersecurity risks into the space along with its advancements [1]. Cybersecurity utilizes technologies along with practices and processes to safeguard systems while protecting networks and data from digital intruders who commit various attacks, including unauthorized system entry and targeted service interferences. Unprecedented cyber-attacks that grow increasingly sophisticated occur more often, thus threatening individual safety along with organizational stability and industrial operations [2]. The developing dangers demand sophisticated, proactive, smart defensive strategies because these present the critical necessity to defend against these evolving threats

Digital system vulnerabilities form the basis for cyber threats that keep becoming trickier to manage with conventional security methods [3][4][5]. The necessity for cybersecurity systems that can quickly identify, halt, and defend against security threats during active attacks makes them more adaptable. The attacks create devastating effects that include monetary loss, together with harm to reputation, and threaten public safety [6][7][8]. Elements of advanced defense mechanism development become essential for businesses and organizations to counter the increase of sophisticated cyber threats and pre-emptively safeguard against new attacks before damage occurs.

The risk of cyberattacks exists primarily against Industrial Control Systems (ICS) since these systems control and automate vital industrial operations [9][10]. Industrial organizations utilize these systems to control their key processes in manufacturing industries and both energy generation facilities and transportation systems. Multiple cyberattacks threaten Industrial Control Systems because organizations continue to increase their dependence on internet access and cyber-enabled applications [11][12][13]. ICS vulnerability to cyber-attacks includes potential operation disruptions along with monetary losses and increased safety risks to the public. Defending ICS infrastructure remains essential for industrial operators and is essential to secure both national and worldwide infrastructure systems

*Corresponding author's ORCID ID: 0000-0000-0000-0000
DOI: https://doi.org/10.14741/ijcet/v.13.6.11

In order to safeguard ICS against expanding cyberthreats, automated threat identification and risk mitigation techniques are becoming more and more important. The detection methods of signature-based detection coupled with manual monitoring prove inadequate for modern changing security conditions. The capacity of ML and DL to analyse enormous volumes of real-time data and identify complex warning patterns of cyberthreats results in effective solutions [14]. ML and DL technology applied to ICS systems enables immediate, better threat identification that reduces organization response times and shields infrastructure from cyber-attack damage.

Motivation and Contribution of the Study

The study addresses the rising cybersecurity threats against industrial control systems, particularly those important for industrial infrastructure. A cyberattack on ICS systems causes major operational disruptions and financial losses, together with increased security risks. Traditional security has limits in terms of scalability and cannot identify threats rapidly, despite its positive outcomes. The research develops improved DL methods to enhance threat detection along with risk reduction in ICS environments because they provide increased accuracy and efficiency. The following is a list of this study's primary contributions: Utilization of the CICIDS-2017 dataset for cyber threat industrial control systems detection.

Preprocess the data to remove inconsistencies and handle missing values.

To determine which characteristics were most important for model performance, feature selection was done. standardized data using Z-score normalisation, rescaling features to have a standard deviation of 1 and a mean of 0.

Employed a CNN as the primary classification model, using convolutional layers.

Model performance was assessed using F1-score, AUC-ROC, recall, accuracy, and precision.

Justification and Novelty

The study is justified by the fact that cyberthreats are becoming more complicated and frequent, which calls for more advanced and dependable detection systems than traditional ones. By leveraging the CICIDS-2017 dataset and employing DL, particularly a CNN, this research introduces an effective approach to automatically extract hierarchical features for improved threat detection. The novelty of the study stems from its integrated pre-processing pipeline, careful feature selection, and the application of a CNN model typically used in image processing adapted for cybersecurity, demonstrating enhanced performance in detecting sophisticated attack patterns.

Structure of the paper

The following structure of the paper is as follows: Section II provides the background study on cyber threat detection. Research approach for this study is provided in Section III. Section IV provides the experiment's findings and a performance analysis of the model. Section V offers the study's conclusion and next directions.

Literature Review

In this section, the study reviews the literature on threat classification and detection. The vast bulk of the reviewed literature focused on classification techniques.

Bhure et al. (2022) Detecting and avoiding fake components has become a top issue, and several techniques have been developed to assess the ICs' validity. Many data sets, processing power, and time are needed to train machine learning-based models. With limited resources, the suggested automated model uses a transfer learning approach to detect counterfeit ICs from several picture capture modalities, yielding more accurate findings. A comparative analysis shows that VGG16 produces a prediction with 80% accuracy that is both resilient and generalized. In addition to the Inception v3 model, the proposed method uses a number of pre-trained models, such as the VGG16 and VGG19 vision models [15].

Wang et al. (2022) Cyberattacks on key infrastructures and modern industrial systems are becoming more frequent, and if they are not identified quickly, they can cause serious operational and financial harm. Seven criteria are used to evaluate the models using actual datasets from gas pipeline and water storage tank systems (e.g., accuracy, F1-score, AUC). Because of its robustness, overfitting avoidance, and feature invariance, XGBoost performs better than other methods, making it an effective way to detect cyberattacks in industrial networks [16].

Mubarak et al. (2021) The ICS test kit produced industrial datasets that include the industrial processes' cyber-physical system. These datasets, which comprise a typical baseline and several industrial hacking situations, are analyzed for research purposes. Metadata is obtained by deep packet inspection (DPI) of network packet flow characteristics. DPI analysis allows for greater understanding of the contents of OT data, contingent on communication protocols. After the industrial datasets have been profiled and pre-processed, DPI is utilized to examine the anomalies. The processed data is normalized to facilitate algorithm analysis, and it is then modelled for anomaly identification using ML-based, cutting-edge DL ensemble LSTM algorithms. These days, the deep learning method is employed to improve OT IDS performance [17].

Dutta and Kant (2021) The combination of combining ML methods and the Cyber Threat Intelligence (CTI) platform with conventional protection measures helps us create a secure, reliable, and efficient structure for clever gadgets to address all present and future security issues. It also aids in the

development of an automated, adaptable security architecture for IoT devices. With the help of the TensorFlow module and the CTI platform, it created a TinyML-based framework that uses an NB supervised ML classifier to anticipate potential dangers that can infect smart gadgets. The end result is a threat prediction accuracy of 96.8% and 96.3% for the training and test datasets. Modern cybercriminals are using improved TTPs to undermine the conventional signature-based threat detection method [18].

Bulle et al. (2020) assesses SCADA communication over time at the OS level, identifies and selects the best operating system to employ for intrusion detection opportunistically for dependability. Experiments conducted using the front-end of several SCADA operating systems demonstrate that OS diversity increases detection accuracy by up to eight more attack types and expands the scope of intrusion detection. Furthermore, their idea may opportunistically select the most reliable OS to use for the current environment behavior, improving the system accuracy by up to 8% on average, in contrast to a single OS method [19].

Table I, Limits, and Future Work presents a comparative overview of the background research based on its findings.

Table 1 Comparative Analysis of Machine Learning Approaches for Cyber Threat Detection

Author	Source	Methodology	Findings	Limitation	Future Work
Bhure et al. (2022)	Detection of counterfeit ICs	Transfer learning with limited resources using pre-trained models (VGG16, VGG19, Inception v3)	VGG16 model achieved 80% accuracy, offering robust and generalized predictions for detecting counterfeit ICs	Requires image data and may not generalize to all types of ICs	Apply the model to broader IC types and real-time industrial inspection systems
Wang et al. (2022)	Cybersecurity in industrial systems	Used real datasets (water storage, gas pipelines); evaluated with 7 metrics; XGBoost for detection	XGBoost fared better than the others because of its feature invariance, robustness, and resistance to overfitting.	May be dataset-specific; needs validation across other industrial setups	Expand to additional industrial domains and incorporate hybrid models
Mubarak et al. (2021)	OT traffic anomaly detection	Deep Packet Inspection (DPI) + metadata profiling + LSTM ensemble deep learning model	Enhanced detection of industrial cyberattacks through rich OT traffic metadata and deep learning	High computational cost and dependency on DPI tools	Optimize for real-time performance and reduce model complexity
Dutta and Kant (2021)	IoT cybersecurity	CTI + TinyML framework using Naive Bayes (NB) in TensorFlow for threat prediction	Achieved high accuracy (96.8% training, 96.3% test) for IoT threat prediction	TinyML-based models may have limitations in adaptability across different IoT devices	Develop a more dynamic CTI-TinyML hybrid framework for evolving TTPs
Bulle et al. (2020)	SCADA OS-level intrusion detection	OS-level analysis of SCADA communications; evaluates multiple OS front-ends for intrusion detection	OS diversity improves detection accuracy, adding 8 new attack categories and increasing system accuracy by 8%	Limited to SCADA environments and OS configurations	Broaden the approach to heterogeneous ICS environments and explore adaptive OS switching strategies

Methodology

This study aims to assess DL and ML models for cyber threat identification. The CICIDS-2017 dataset is used in the suggested approach for cyber threat detection. Then, data pre-processing was performed, which included handling missing values using imputation techniques, removing inconsistencies and duplicates, and applying one-hot encoding for categorical features to convert them into numerical representations. After selecting the most significant features using feature selection, the data was standardised using Z-score normalisation, which rescaled the characteristics must have a standard deviation of one and a mean of zero. Twenty percent was set aside for testing, while 80 percent of the pre-processed data was kept for training. The primary classification model employed was a CNN with convolutional layers. The model's efficacy was evaluated using F1-score and AUC-ROC metrics, as well as accuracy tests combined with precision and recall.

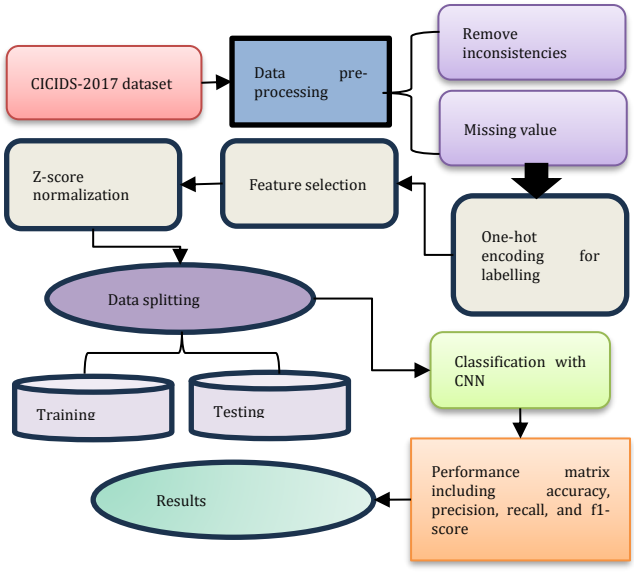

Fig.1 Flowchart of Cyber Threat Detection

The entire process of the Cyber threat detection Figure 1 flowchart appears in the diagram below:

Data Collection

The CICIDS2017 dataset exists as a complete benchmark dataset specifically developed for IDS applications. Its 50,000 samples that contain 80 features that describe network traffic characteristics. This dataset comprises two distinct categories of instances, where 25,000 entries belong to the normal class and 25,000 instances represent anomalies. The dataset functions as the essential base to build and validate cyber threat detection systems. The tools for visual representation that include heatmaps and pie plots show feature distribution patterns and attack patterns through graphs as demonstrated in below:

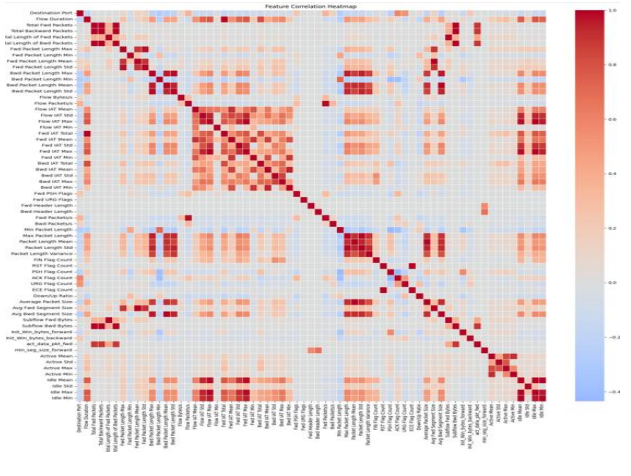

Fig.2 Heatmap Matrix for Various Features

A heatmap contained in Figure 2 depicts the various feature correlations in the dataset through negative blues and positive red hues. Strong positive and negative feature correlations show themselves through darker red and blue square cells, respectively. Self-correlation produces a line that runs diagonally across the heatmap because it has a constant value of 1, while the heatmap displays symmetry about this axis. The many weak correlations shown in light color help uncover relationships between features, which experts can use to better defend their cybersecurity systems.

Data Preprocessing

The processing of ML data requires organizers to change disorderly, unprocessed information into predictable structures that models can utilize. This step is essential because raw data often contains missing values, inconsistencies, and redundancies. The pre-processing actions described below are as follows:

Remove inconsistencies: Data cleaning in threat detection removes inconsistencies, duplicates, and irrelevant data while preserving critical threat-related patterns. This enhances ML model accuracy, enhancing real-time threat detection and lowering false positives.

Missing value: In cyber threat detection, missing values can distort are handled using imputation techniques to maintain data integrity and improve model accuracy.

One-Hot Encoding for Labelling

Data encoding refers to converting the numerical representation of categorical data that ML systems may utilize. One-hot encoding is basically a feature engineering method for nominal categorical data. Applying machine learning (ML) to categorical data without a tree-based approach requires that the data be converted into numerical form.

Feature Selection

Finding the characteristics that have the most influence on an issue is known as feature selection. To find the most pertinent characteristics, feature selection is utilized and is often used due to its performance-enhancing properties. Determining which attributes should be included requires a deep comprehension of the facts being utilized. The score of importance features is provided in below:

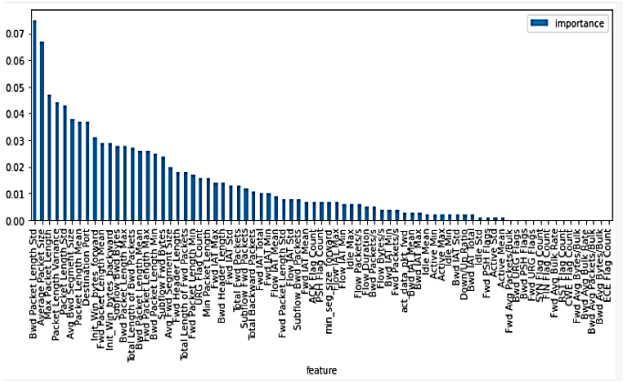

Fig.3 Importance score of Each Feature

Figure 3 shows a bar graph showing each feature's importance score from the CIC-IDS2017 dataset. With relevance ratings ranging from 0.00 to 0.07, it draws attention to the significance of different elements within the dataset. The features on the left side of the graph have higher importance scores, indicating they are more influential in the context of the dataset.

Z-Score Normalization

The z-score normalization is one of the most popular and effective normalization techniques. Z-score normalization helps handle features with different scales (e.g., packet size, and flow duration) to enhance machine learning models' performance. This ensures that no single feature dominates. Z-score normalization transforms data by rescaling characteristics with a standard deviation of Equation (1) and a mean of 0.

$$Z = \frac{X-\mu}{\sigma} \quad (1)$$

Where μ is the feature mean and X is the original data point, σs the standard deviation.

Data Splitting

Two pre-processed data sets are available: one for testing and one for training. The remaining 80% of the data is used in the testing set to evaluate the model's performance, while the remaining 20% is utilized in the training set to train the model.

Classification With CNN Model

The first CNN block uses a convolutional layer, then Relu activation, max pooling, and dropout to efficiently extract and regularise features. The CNN blocks use varying kernel sizes to extract diverse features, followed by a flattened output passed to an MLP with dense layers, L2 regularization, batch normalization, Relu activation, and dropout to reduce overfitting and improve learning [20]. The final network layer uses the proper cross-entropy loss function for optimization, sigmoid for multi-class classification using SoftMax, and for binary classification using [21]. The input data categories are chosen by the attributes. By filtering the input data, the convolution operation in CNN layers serves as a crucial step in feature extraction. By calculating the dot product at each sliding position, the convolution kernel's sliding movement over the feature map creates a new feature map. Equation (2) provides a description of the mathematical formulation of convolution processes.

$$z_{i,j} = (X * K)_{i,j} = \sum_m \sum_n Z_{i+m,j+n}\, k_{m,n} \quad (2)$$

The ReLU activation function is a straightforward yet effective nonlinear transformation that is displayed in Equation (3). This method improves training speed and system performance by sparsely activating components, maintaining positive values, preventing gradients from disappearing, and producing zero outputs for negative inputs.

$$ReLU(x) = max\,(0, x) \quad (3)$$

This max pooling method lowers the input map spatial size but maintains key information within it. As a part of the operation it extracts the largest value present in a defined pooling area to achieve input reduction. Equation (4) presents the mathematical representation of max pooling operation.

$$p_{i,j} = \max\,(X_{i:i+p,j:j+q} \quad (4)$$

One training technique involves the dropout layer which enables random zero-filling of p percentage input units to prevent model overfitting. The application of this method improves model generalization because it prevents dependence on individual input features according to Equation (5).

$$Dropout(x) = \begin{cases} x \text{ with probability } 1-p \\ 0 \text{ with probability } p \end{cases} \quad (5)$$

The final output layer applies sigmoid activation to produce probability values that show the chance of instances belonging to the positive category. The mathematical representation of this appears in the Equation (6).

$$\sigma(z) = \frac{1}{1+e^{-z}} \quad (6)$$

The last dense layer produces Z which represents the output value. The output layer performs multi-class categorization using the SoftMax algorithm. Producing probability distributions across different classes through the model. This may be expressed quantitatively using Equation (7).

$$softmax(z_i) = \frac{e^{z_i}}{\sum_j e^{z_j}} \quad (7)$$

where z_i is the raw score for class j is represented by the output for class i and z_j.

Performance Metrics

A collection of assessment criteria, sometimes referred to as performance metrics, was employed to assess the efficiency of detecting cyber threats. A table illustrating the extent to which A classification model, also referred to as a "classifier," operates as a confusion matrix describes when it is applied to a set of test data for which the true values are known. The ML model's predicted values and the actual target values are contrasted in the matrix[22]. The final models were evaluated using Five evaluation measures were employed to evaluate the final models: F1-score, recall, accuracy, and precision. TP, FP, TN, and FN are the first metrics used by confusion matrices to assess the models:

True positive (TP): An assault sample has been appropriately classified as such.

True negative (TN): It has been accurately determined that a normal sample represents typical traffic.

False positive (FP): An attack has been incorrectly identified from a typical sample.

False negative (FN): A sample of an assault was mistakenly classified as regular traffic.

Accuracy: The proportion of all samples with correctly recognized classes. In balanced datasets, this statistic is commonly used to evaluate the efficacy of an IDS. It is expressed in Equation (8):

$$Accuracy = \frac{\text{TN} + \text{TP}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \quad (8)$$

Precision: shows how many attack samples were completely predicted out of all the expected attack samples. It is represented as Equation (9):

$$\text{Precision} = \frac{\text{TP}}{\text{TP+FP}} \quad (9)$$

Recall: Recall, also known as sensitivity, is the proportion of correctly predicted attack samples to all samples that make up an attack. This measurement is occasionally called the Detection. It is represented in Equation (10):

$$\text{Recall} = \frac{\text{TP}}{\text{TP+FN}} \quad (10)$$

F1-score: The harmonic means of the precision and recall parameters. The F1 score provides improved system assessment by presenting the gap between Precision and Recall to determine solution balance. The definition of F1 score consists of the following description Equation (11):

$$\text{F1} = 2 * \frac{(\text{precision} * \text{recall})}{\text{precision+recall}} (11)$$

AUC-ROC: AUC serves as the main evaluation tool for the model through its ROC curve's area under one measurement. The model output was sorted by prediction result, and the samples were labelled positive one after another. The TPR and FPR values were calculated through a two-axis system where TPR went across and FPR went down according to Equations (12) and (13):

$$TPR = \frac{TP}{TP+FN} (12)$$
$$FPR = \frac{FP}{TP+FP} (13)$$

The AUC value serves as an important measure to assess a model which ranges from 0.5 to 1 with high scores indicating strong generalization and classification accuracy.

Results and Discussion

The project used Python v3.10 on a powerful system having an Intel i7 12th-generation processor with 16GB RAM memory and a 512GB SSD storage device with the graphics chip variant of 1050 H to build and test the cyber threat detection model. The results of the several categorization techniques utilized in this study for cyber threat detection are examined in this part. Using the CICIDS-2017 dataset, the CNN-based suggested model's performance is contrasted with that of KNN, NB, and SVM. The models were assessed using key performance indicators such as F1-score, recall, accuracy, and precision. Table II shows the efficacy of the proposed model. With a F1-score, recall, accuracy, and precision of 99.58%, the CNN exhibits remarkable performance qualities. These outcomes demonstrate how strong and dependable the CNN is for the classification task.

Table 2 Experiment Results of Proposed CNN for Cyber Threat Detection

Performance Matrix	Convolutional Neural Network (CNN)
Accuracy	99.58
precision	99.58
Recall	99.58
F1-score	99.58

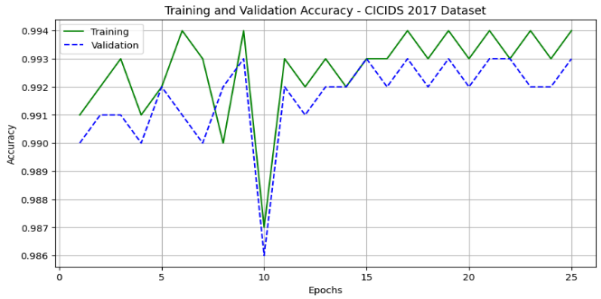

Fig.4 Accuracy Graph for CNN

The precision of a CNN's validation and training is seen in Figure 4 on the CICIDS 2017 dataset over 25 epochs. Both accuracies start around 99% and steadily improve, with a brief dip around epoch 9–10 before quickly recovering. The accuracy of training is somewhat higher than that of validation, but both remain closely aligned between 99–99.4%, indicating effective learning with minimal overfitting.

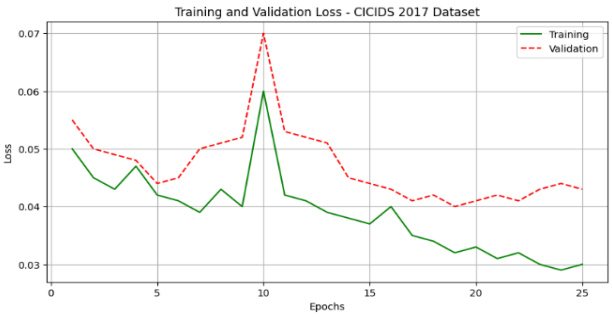

Fig.5 Loss Graph for CNN

On the CICIDS 2017 dataset, Figure 5 shows training and validation loss of the CNN during 25 epochs. While validation loss is greater and more erratic, peaking around epoch 10 before stabilizing around 0.04, training loss gradually drops from around 0.05 to about 0.03. The divergence between the two indicates mild overfitting, despite overall improvement and high accuracy.

Figure 6's confusion matrix compares real and expected labels to show categorization performance. While the columns indicate the predicted classes. The CNN model identified 5,005 anomaly cases and 4,934 normal cases properly. Nevertheless, it incorrectly identified 22 anomalous cases as normal FN and 39 normal cases as anomalies FP.

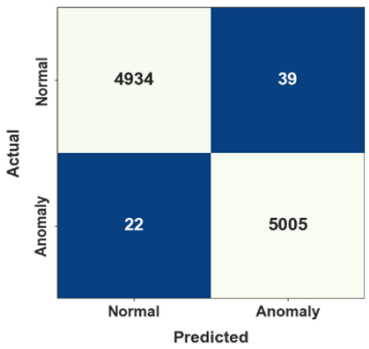

Fig.6 Confusion Matrix for CNN

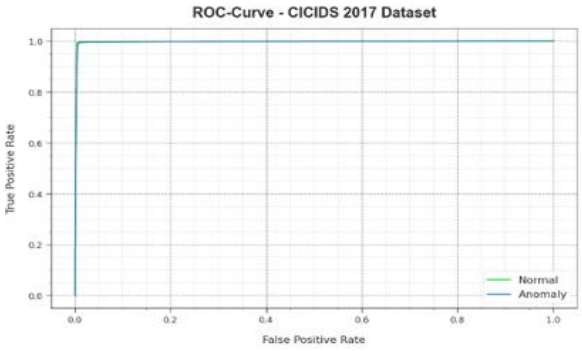

Fig.7 ROC curve of CNN

The CNN model's ROC curve, which was evaluated using the CICIDS 2017 dataset, is then displayed in Figure 7. The FPR is displayed on the x-axis from 0 to 1, while the TPR is shown on the y-axis from 0 to 1. Both the normal and anomalous classes' ROC curves.

Table 3 Comparative analysis for cyber threat detection between existing models' performance

Performance Matrix	Proposed	Comparison model	
	CNN	***SVM****[23]*	***KNN****[24]*
Accuracy	99.58	96.98	94
precision	99.58	95.78	86
Recall	99.58	98.30	90
F1-score	99.58	97.02	88

Table III above presents a comparison of model performance. It was seen that CNN reached the highest accuracy of 99.58% in this comparison, which is greater than other algorithms such as SVM 96.98%, KNN 94% and NB 72.96%. The CNN also achieved exceptionally high precision, 99.58%, and recall, 99.58%, which demonstrated the CNN's capability to minimize and maximize actual positive detections while minimizing false positives. Additionally, SVM and KNN also scored well in comparison, with SVM attaining a recall of 98.30% and a 95.78% accuracy and a somewhat lower 90% recall for KNN. NB was able to recall 96.71% of the positive cases, but with a lower precision of 65.76%, it cannot be assumed that it will be able to find all positive cases. In general, CNN performs better than every other model. in terms of all the metrics, making it the best in the case of Cyber threat detection.

Conclusion and Future Work

Cyber threats against Industrial Control Systems keep becoming more advanced, which means organizations need better ways to find and defend against attacks. The research showed that DL technology improves security systems at ICS facilities. This study shows that DL technology, especially CNN, can find cyber dangers using the CICIDS-2017 dataset. The CNN model consistently outperforms other models in traffic categorization, with F1-score, recall, accuracy, and precision scores of 99.58%. The graphs display proper model training and show small amounts of overfitting problems with strong predictive output results. This study faces three key constraints related to the detection system, including its weakness against unknown attacks, demanding processing needs, and limited protection for all types of threats. The next research needs to include more network security data and test multiple DL methods, including RNNS and transformers, to create faster and more reliable systems that find new hacking patterns automatically.

References

[1] D. Bhamare, M. Zolanvari, A. Erbad, R. Jain, K. Khan, and N. Meskin, "Cybersecurity for Industrial Control Systems: A Survey," *Comput. Secur.*, vol. 89, no. 1, p. 101677, Feb. 2020, doi: 10.1016/j.cose.2019.101677.

[2] A. Balasubramanian, "Building Secure Cybersecurity Infrastructure: Integrating AI and Hardware for Real-Time Threat Analysis," *Int. J. Core Eng. Manag.*, vol. 6, no. 07, pp. 263–271, 2020.

[3] V. Kolluri, "A Thorough Examination of Fortifying Cyber Defenses : AI in Real Time Driving Cyber Defence Strategies Today," *Int. J. Emerg. Technol. Innov. Res.*, 2018.

[4] S. S. S. Neeli, "Key Challenges and Strategies in Managing Databases for Data Science and Machine Learning," *Int. J. Lead. Res. Publ.*, vol. 2, no. 3, p. 9, 2021.

[5] V. Kolluri, "A Pioneering Approach To Forensic Insights: Utilization AI for Cybersecurity Incident Investigations," *Int. J. Res. Anal. Rev.*, vol. 3, no. 3, 2016.

[6] V. Pillai, "Anomaly Detection for Innovators: Transforming Data into Breakthroughs," *Lib. Media Priv. Ltd.*, 2022.

[7] S. Singamsetty, "Fuzzy-Optimized Lightweight Cyber-Attack Detection For Secure Edge-Based IoT Networks," *J. Crit. Rev.*, vol. 6, no. 7, 2019, doi: 10.53555/jcr.v6:i7.13156.

[8] V. Kolluri, "A Detailed Analysis of AI as a Double-Edged Sword: AI-Enhanced Cyber Threats Understanding and Mitigation," *Int. J. Creat. Res. Thoughts*, vol. 8, no. 7, 2020.

[9] P. Pathak, A. Shrivastava, and S. Gupta, "A survey on various security issues in delay tolerant networks," *J Adv Shell Program.*, vol. 2, no. 2, pp. 12–18, 2015.

[10] S. Pandya, "Innovative blockchain solutions for enhanced security and verifiability of academic credentials," *IJSRA*, vol. 06, no. 01, pp. 347–357, 2022.

[11] G. Sakellariou, P. Fouliras, I. Mavridis, and P. Sarigiannidis, "A Reference Model for Cyber Threat Intelligence (CTI) Systems," *Electronics*, vol. 11, no. 9, 2022, doi: 10.3390/electronics11091401.

[12] S. Chatterjee, "Mitigating Supply Chain Malware Risks in Operational Technology : Challenges and Solutions for the Oil and Gas Industry," *J. Adv. Dev. Res.*, vol. 12, no. 2, pp. 1–12, 2021.

[13] V. Kolluri, "Cutting-Edge Insights into Unmasking Malware: AI-Powered Analysis and Detection Techniques," *JETIR - Int. J. Emerg. Technol. Innov. Res.*, vol. 4, no. 2, p. 33, 2017.

[14] K. K. Nimavat and R. Kumar, "Updating Machine Learning Training Data Using Graphical Inputs," 17178360, 2022

[15] C. M. Bhure, G. S. Nicholas, S. Ghosh, Y. Zhong, and F. Saqib, "Automated Transfer Learning Model for Counterfeit IC Detection," in *2022 IEEE Physical Assurance and Inspection of Electronics, PAINE 2022*, 2022. doi: 10.1109/PAINE56030.2022.10014980.

[16] W. Wang, F. Harrou, B. Bouyeddou, S.-M. Senouci, and Y. Sun, "Cyber-Attacks Detection in Industrial Systems Using Artificial Intelligence-Driven Methods," *Int. J. Crit. Infrastruct. Prot.*, vol. 38, Sep. 2022, doi: 10.1016/j.ijcip.2022.100542.

[17] S. Mubarak, M. H. Habaebi, M. R. Islam, and S. Khan, "ICS Cyber Attack Detection with Ensemble Machine Learning and DPI using Cyber-kit Datasets," in *Proceedings of the 8th International Conference on Computer and Communication Engineering, ICCCE 2021*, 2021. doi: 10.1109/ICCCE50029.2021.9467162.

[18] A. Dutta and S. Kant, "Implementation of Cyber Threat Intelligence Platform on Internet of Things (IoT) Using TinyML Approach for Deceiving Cyber Invasion," in *International Conference on Electrical, Computer, Communications and Mechatronics Engineering, ICECCME 2021*, 2021. doi: 10.1109/ICECCME52200.2021.9590959.

[19] B. B. Bulle, A. O. Santin, E. K. Viegas, and R. R. Dos Santos, "A Host-based Intrusion Detection Model Based on OS Diversity for SCADA," in *IECON Proceedings (Industrial Electronics Conference)*, 2020. doi: 10.1109/IECON43393.2020.9255062.

[20] R. Tarafdar and Y. Han, "Finding Majority for Integer Elements," *J. Comput. Sci. Coll.*, vol. 33, no. 5, pp. 187–191, 2018.

[21] A. Tekerek, "A novel architecture for web-based attack detection using convolutional neural network," *Comput. Secur.*, 2021, doi: 10.1016/j.cose.2020.102096.

[22] P. Vanin *et al.*, "A Study of Network Intrusion Detection Systems Using Artificial Intelligence/Machine Learning," *Applied Sciences (Switzerland)*. 2022. doi: 10.3390/app122211752.

[23] A. Rosay, E. Cheval, F. Carlier, and P. Leroux, "Network Intrusion Detection: A Comprehensive Analysis of CIC-IDS2017," in *International Conference on Information Systems Security and Privacy*, 2022. doi: 10.5220/0010774000003120.

[24] A. Ferriyan, A. H. Thamrin, K. Takeda, and J. Murai, "Generating Network Intrusion Detection Dataset Based on Real and Encrypted Synthetic Attack Traffic," *Appl. Sci.*, vol. 11, no. 17, 2021, doi: 10.3390/app11177868.