DOI: http://dx.doi.org/10.26483/ijarcs.v16i6.7387

Volume 16, No. 6, November-December 2025

ISSN No. 0976-5697

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

PREDICTIVE THREAT MODELING IN INDUSTRIAL IOT (IIOT) NETWORKS USING MACHINE LEARNING TECHNIQUES IN CLOUD ENVIRONMENTS

Ram Pratap Singh
Department of Computer Science and Engineering
Lakshmi Narain College of Technology
Bhopal

Abstract—Cloud-enabled Industrial Internet of Things (IIoT) networks are being used more and more. These networks have changed how automation, tracking, and control are done in factories, but they have also made security much harder because they create so much different and changing data. There is an absolute need for predictive threat modeling in IIoT cloud environments because traditional signature-based solutions are notoriously bad at detecting new and unknown assaults. In order to forecast potential threats in IIoT network traffic, this research suggests a machine learning-based strategy, using the CICIDS 2017 dataset as a baseline. After extensive preprocessing operations such as data cleaning, normalization, feature selection, and data balancing through SMOTE, a Convolutional Neural Network (CNN) was trained to automatically draw and understand complicated spatial-temporal patterns from multidimensional traffic data. Accuracy (ACC) was 99.23%, precision (PRE) was 98.32%, recall (REC) was 99.15%, and F1-score (F1) was 98.35%; this model outperformed its competitors, which included Logistic Regression (84.1%), LSTM (93.78%), and MLP (97.7%). proving that it can separate legitimate traffic from malicious ones. Findings show that predictive threat modeling based on deep learning is a good way to make IIoT networks in the cloud more secure and reliable.

Keywords—CICIDS-2017 Dataset, Internet of Things, Intrusion Detection, Industrial Internet Of Things (IIoT), Security, Cyber-Attacks, Machine Learning, CNN.

I. INTRODUCTION

Cloud computing (CC) has become a significant aspect of current digital infrastructure because it offers flexible, on-demand storage, computation, and analytics tools [1][2]. Its ability to centralize large-scale data processing while supporting distributed access has enabled enterprises to implement real-time monitoring, decision-making, and resource optimization across multiple sectors. Its widespread adoption in recent years can be attributed to its capacity to offer adaptable infrastructure, resources available on demand, and worldwide connectivity [3][4][5] The flexibility and computational power of cloud platforms make them particularly suitable for managing the growing data volumes generated by connected devices, laying the foundation for the next generation of industrial digitalization.

The IIoT is based on cloud capabilities that involve the integration of sensors, actuators, and controllers into industrial systems to measure, transmit, and process the operational data [6][7][8]. By putting physical equipment connected to digital networks, the IoT enables the automation, predictive maintenance, as well as optimization of its performance. Nevertheless, the growing interconnection of devices also raises the cyber-attack area, and thus the industrial networks become vulnerable to attacks [9][10][11]. This underscores the importance of powerful policies that could be used to utilize cloud infrastructure to track and protect IIoT ecosystems [12][13][14].

Predictive threat modelling with the integration of IIoT and cloud computing, there are predictive indications in advance of the possible vulnerabilities and the reduction of the risks prior to them affecting the operations [15][16][17]. By consolidating device-generated data in the cloud, predictive models may be used to detect anomalies, forecast attacks, as well as prioritize interventions. The cloud environment provides the computational resources required to improve such models on a repetitive basis, which provides the ability to adjust to the new challenges arising in the real-time and ensures the stability of the industrial operations.

The predictive threat modelling using AI and ML methods can help learn complex patterns in IIoT data on clouds automatically [18][19][20]. The application of ML techniques is appropriate due to their flexibility; they can be applied in the categorization of data, clustering, and the detection of anomalies (AD) among other applications. The most efficient and basic approach to combating cyberattacks has become ML. Supervised, unsupervised, and DL algorithms all are able to predict cyber incidents and classify anomalous behavior and recommend preventative measures [21][22][23]. Integrating AI/ML with cloud-based IIoT networks does not only enhance security but also helps with intelligent automation, ongoing system enhancement, and effective use of resources, providing an effective framework of a secure and smart industrial work.

A. Motivation and Contribution

This study is fuelled by the increasing sophistication and rate of cyber-attacks, which threaten not only the network security but also the integrity of the data. The lack of ability of traditional security mechanisms to identify advanced or dynamic attack patterns exposes systems to attack. The aim of the project is to come up with a smart and automated system that can effectively detect existing and new threats with a high degree of reliability using DL methods, that is, CNNs. The ultimate outcome of the attempts to make the network intrusion detection systems more effective, reliable, and resilient is improved security of the vital digital infrastructures and private data. As research, this study has made a number of important contributions as follows:

- Use of the CICIDS 2017 data, which used a realistic and all-inclusive dataset of various attack categories and normal traffic to provide sound model testing.

- Cleaning, normalization (z-score), feature selection, outlier removal and label encoding to improve model ACC are included.
- Addressing class imbalances and maintaining proportional class distribution in training and testing sets to improve generalization.
- Using bar plots, heatmaps, and class distribution charts to understand attack patterns and feature correlations.
- Development of a CNN-based predictive threat detection model, leveraging deep learning to automatically learn complex patterns from network traffic data.
- Delivering a dependable, scalable, and strong solution for identifying malicious and benign traffic in actual network settings.
- The model was rigorously evaluated using a variety of performance measures. F1, REC, ACC, and PRE were among these measurements.

B. Structure of the Paper

The Structure of the paper is as follows: In Section II, delve into previous research on predictive threat modelling. In Section III, present the dataset, pre-processing techniques, and model implementation. In Section IV, analyze and compare the experimental results. Finally, in Section V, summarize the main findings and suggest areas for future research.

II. LITERATURE REVIEW

The development of this research was informed and supported by a comprehensive examination and analysis of leading research works dedicated to predictive threat modelling in IIoT in cloud environment.

Haider et al. (2025) suggested a new paradigm that is able to integrate GANs with transfer learning to enable creation of truer to life detection systems that are able to detect even the more hidden versions of hacks which are still unknown. The collection of data includes assembled huge amounts of heterogeneous data, both past attack data, network traffic records, UNSW-NB15, and CICIDS data, and the data acquired by cloud service providers. It is followed by a new method that combines both the steps, the first step would be the synthesis of synthetic data by GANs and the second step would be the enhancement of the feature representation of the same. The assessments indicate that the level of the hybrid model usage has proven to possess an incredible PRE of 98% in regard to the ability to detect both new and traditional forms of attacks [24].

Ch et al. (2025) Industrial IoT (IIoT) environments through assessing the efficacy of diverse machine learning methods with regard to enhanced threat detection and intrusion prevention. This paper compares the operation of various algorithms. The characteristics of these algorithms were tested in terms of processing times and ACC in terms of training and testing. DT Algorithm proved to be the most effective and had a high-test ACC. LR and RF also showed significant results with accuracies of 98% each, the possibility to utilize machine learning models as the means of improving the security of the IIoT by quickly identifying and addressing the possible cyber threats and, therefore, allowing the deployment of industrial solutions that are more secure and reliable [25].

Mohammed et al. (2025) outlines an innovative Anomaly Detection System created to be utilized in Secure Cloud Computing Environments that utilizes machine learning techniques. It makes use of a Stacking Ensemble Model, which combines the advantages of several distinct models to improve performance and ACC. Training base learners on the updated NSL-KDD+ dataset makes use of five distinct ML algorithms DT, KNN, SVM, RF, and Gradient Boosting Classifier. LR, a last meta-learner, learns to integrate the predictions of all these models into a final classification using the inputs from the previous ones. With a remarkable ACC of 99.99%, the intended Stacking Ensemble Model outperforms previous studies. This approach is effective in detecting anomalies and reducing possible dangers in cloud computing settings, as demonstrated by the considerable improvement in ACC [26].

Houkan et al. (2024) industrial IoT networks by simplifying and selecting features. The effectiveness of two approaches—Minimum Redundancy Maximum Relevance and Principal Component Analysis for feature selection and reduction was examined. Algorithms for DNN networks, SVM, DT, and KNN are a few concepts used in this area. Minimum Reliability Maximum Relevance can achieve high ACC with as few as 12 features, but KNN achieved 99% ACC in multi-class classification. In this look at how to apply MRMR and PCA algorithms to different feature sets, and focus on feature selection and reduction as part of feature engineering. In comparing these approaches, they demonstrate their power in performance and the complexity of the model [27].

Vinolia, Kanya and Rajavarman (2023) The IDS is an element used in securing and compliance procedures in computer systems that safeguards cloud environments against a wide range of threats and attacks. This article aims at researching the application of DL and ML networks in different methodologies, through the different phases of intrusion detection, to achieve better results. This work is aimed to comment on the state of the art in the context of intrusion detection with the help of various methods, among them, soft computing, data mining and others are involved. Results show that unsupervised deep learning methods outperform supervised methods with a 99.95% success rate [28].

Hossain and Islam (2023) including scalability and flexibility, with a pay-as-you-go model. However, these features come with some drawbacks along with their benefits. Among the challenges posed by the flexibility and pay-as-you-go attributes of cloud computing, one of the most detrimental threats is Economic Denial of Sustainability (EDoS), leading to significant financial losses for consumers of cloud services. Consequently, an effective and efficient solution is urgently needed to fully realize the benefits of cloud computing. In this research, they aim to address the EDoS attack using machine learning-based methods. research is conducted on the UNSW-NB15 dataset, and the result demonstrates that optimized LightGBM model with chosen hyperparameters surpasses existing models with significant improvements in attack detection, achieving a REC score of 99.21% [29].

Tiwari and Jain (2022) offer a novel DL and ML-based firewall technique for safe cloud computing environments. The suggested approaches employ a special combination mechanism known as most common decision to recognize and classify incoming traffic packets. In order to roughly identify the final assault category, this methodology considers both the ML algorithm's current judgment and the nodes' previous decisions. Additionally, this method improves system ACC and learning efficiency. The results are based on the publicly accessible UNSW-NB-15 dataset. Data indicates that it increases anomaly detection by 97.68% [30].

Table I presents a summary of recent research on predictive threat modelling in cloud environments, highlighting the dataset, methodology, key findings, Limitation and Future work

TABLE I. RECENT STUDIES ON PREDICTIVE THREAT MODELING IIOT IN CLOUD ENVIRONMENTS USING MACHINE LEARNING

Author (Year)	Dataset Used	Methodology	Key Findings	Limitation	Future Work
Haider et al. (2025)	Historical attack data, network traffic logs, UNSW-NB15, CICIDS, cloud datasets	GANs with Transfer Learning (synthetic data + improved feature representation)	Achieved 98% accuracy in detecting both classic and novel attacks	Limited evaluation on real-time streaming data	Extend framework to real-time adaptive detection in large-scale cloud/IIoT
Ch et al. (2025)	IIoT traffic data	Comparison of ML models (DT, LR, RF)	DT achieved highest testing accuracy; LR & RF ~98%	Dataset details unclear, possible generalization issues	Apply on larger IIoT datasets and explore hybrid ML-DL models
Mohammed et al. (2025)	Revised NSL-KDD+	Stacking Ensemble (DT, KNN, SVM, RF, GBC → Logistic Regression as meta-learner)	Achieved 99% accuracy, outperforming prior IDS models	Tested only on benchmark dataset, not on real-world cloud logs	Deployment in real-time cloud with adaptive model updating
Houkan et al. (2024)	IIoT network datasets	Feature engineering with MRMR (selection) & PCA (reduction) + ML models (DT, KNN, G-SVM, NN)	MRMR + KNN achieved 99% accuracy with just 12 features	Focused only on feature reduction/selection, not ensemble learning	Explore DL with optimized feature engineering for IIoT
Vinolia et al. (2023)	Multiple IDS datasets	DL & ML-based intrusion detection (soft computing, data mining, unsupervised DL)	Unsupervised DL achieved 99% accuracy	Lack of benchmark dataset specification	Standardized benchmarking & cross-domain IDS testing
Hossain & Islam (2023)	UNSW-NB15	Optimized LightGBM with hyperparameter tuning	Achieved recall score of 99.21% for EDoS attack detection	Focused only on EDoS, not general IDS	Extend ML solution to broader attack types in cloud
Tiwari & Jain (2022)	UNSW-NB15	Hybrid ML-DL firewall with "Most Frequent Decision" approach	Improved anomaly detection with 97.68% accuracy	Performance tested only on offline dataset	Real-time firewall deployment with adaptive decision fusion

Research gaps: Predictive threat detection in cloud and network systems has progressed significantly, but there are still some unsolved concerns. Existing research mostly uses deep learning, ensemble models, or hybrid approaches to increase ACC; however, these methods are generally inflexible in the face of dynamic, ever-changing threats because they are trained on static or historical information. Furthermore, although there are literature works that cover the multi-class threat detection or even particular attacks, such as DDoS and EDoS, there are still few comprehensive solutions that can be used to identify a wide variety of known and unknown attacks that exist within various cloud-based infrastructures. Complexity, scaling, and real-time deployment are also other challenges that most systems have had to overcome. Additionally, it seems that there are no unified frameworks to combine the robust feature selection, synthetic data generation, and adaptive learning in a dynamic threat environment. These gaps are critical to designing more robust, effective, and generalized predictive threat detection systems in the present-day cloud and network security.

III. RESEARCH METHODOLOGY

A comprehensive approach to threat detection prediction utilizing the CICIDS 2017 dataset is part of the study's methodology, as shown in Fig. 1. Initially, data gathering and analysis were performed, where the dataset containing 2,830,743 rows across eight CSV files with seventy-eight features each was explored, and conceptions such as bar plots, heatmaps, and donut charts were used to examine attack distributions and feature correlations. Pre-processing of data involved addressing missing data, eliminating duplicates and outliers, label encoding categorical data and z-score normalization to normalize the scales of features. Selection of features was done to find out the most relevant variables to model performance and then data balancing was done through the use of SMOTE to handle the class imbalance.

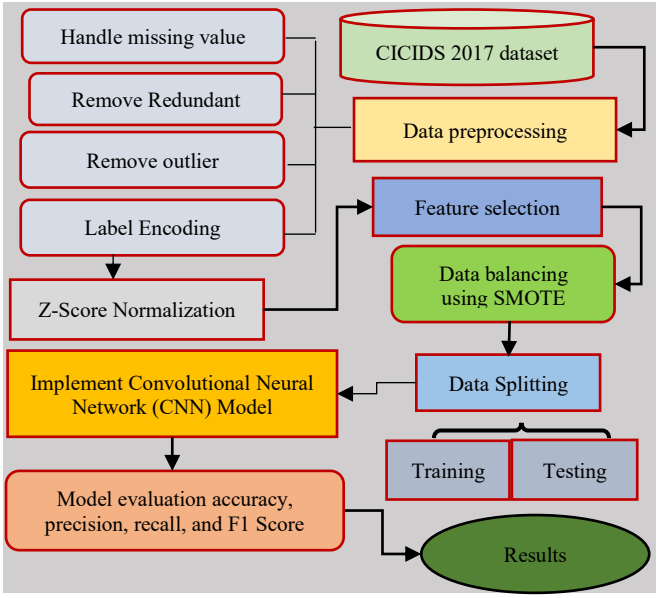

Fig. 1. Proposed Flowchart for Predictive Threat Modeling using Machine Learning

The data was further divided into training and test in 80:20 stratified proportion to maintain the distribution of classes. In the case of predictive modeming, a deep learning CNN was used with the introduction of convolutional layers to obtain hierarchical features of the input data and nonlinear activation functions to create feature maps. A confusion matrix was used to evaluate the model's performance, enabling it to monitor crucial performance parameters like ACC, PRE, REC, and F1, giving a reliable evaluation of the model's ability to discern between malicious and benign data.

The following section presents a detailed explanation of each stage in the proposed methodology:

A. Data Gathering and Analysis

The CICIDS 2017 dataset is used in this investigation. The CICIDS2017 dataset includes twenty-five users' abstract normal activity based on several protocols, including HTTP, HTTPS, FTP, SSH, and email. Additionally, it contains a variety of attack traces, including DDoS, Heartbleed, Web Attack, Infiltration, Botnet, Brute Force FTP, and Brute Force SSH. Eight unique files, each with its own comma-separated values (CSV) format, make up this dataset. Each of the eight files contains 2,830,743 rows, with 78 features and a label of "Benign" or one of fourteen other categories. Following are examples of data visualizations that were employed to investigate attack distribution, feature correlations, etc: bar plots and heatmaps:

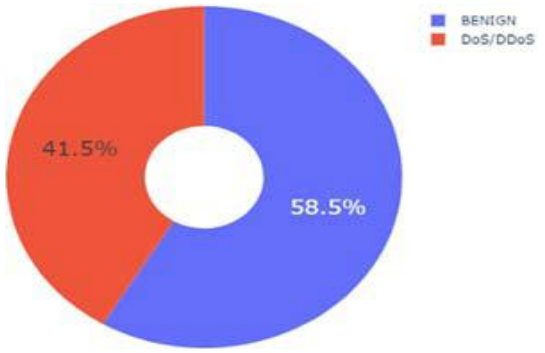

Fig. 2. Frequency Distribution of Class Labels

Fig. 2 The donut chart illustrates the class distribution of the dataset between benign traffic and DoS/DDoS attacks. Benign traffic constitutes the majority with 58.5%, shown in blue, while malicious DoS/DDoS traffic accounts for 41.5%, represented in red reducing bias and improving the robustness of predictive threat detection.

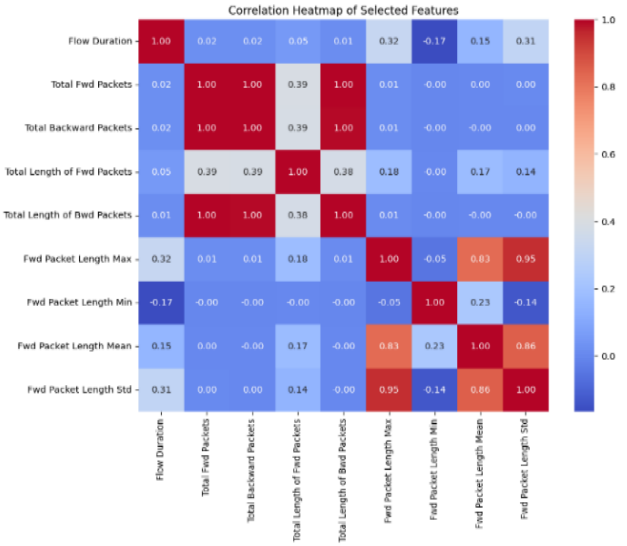

Fig. 3. Correlation Heatmap 0f Selected Feature

This correlation heatmap illustrates Pearson correlation coefficients among nine network flow feature schemes, where red signifies strong positive correlations and blue indicates weak or negative correlations, as shown in Fig. 3. The analyzed features include flow-level parameters (Flow Duration, Total Forward/Backward Packets,) and forward packet length statistical measures. Notable strong correlations exist where volumetric metrics (packet counts and total bytes) correlate with distributional properties collectively enabling comprehensive network performance analysis and traffic pattern recognition.

B. Data Pre-Processing

The CICIDS 2017 dataset was utilized for data preparation, which involved concatenation, cleaning, and feature engineering. The pre-processing steps included managing missing values, eliminating duplicate records and outliers, followed by data labelling and normalization. The key pre-processing procedures are outlined as follows:

- **Handle missing value:** Handling missing values refers to the various statistical and machine learning techniques used to address gaps, blank cells, or nulls in a dataset that arise from incomplete information.
- **Remove Redundant Entries:** To remove redundant entries is the process of identifying and deleting duplicate or identical pieces of data from a dataset to ensure ACC, reduce storage, and improve data management.
- **Remove outliers:** Data points that deviate greatly from the norm are known as outliers, and their removal or handling is known as outlier data cleansing.
- **Label Encoding:** A data preprocessing technique utilized in ML, label encoding transforms categorical variables into a numerical format. Machine learning algorithms often struggle to handle categorical data that is not numerical, therefore this transformation is essential.

C. Normalization using Z-score

Data normalization is a method for transforming and standardizing data so that it follows a specific distribution. Rescaling, z-score normalization, and min-max normalization are the standard methods for normalizing data. Z-score normalization, which has a mean of 0 and a standard deviation of 1, was used in this instance to standardize the data. The values that are closest to the average with a standard deviation of one are transformed by this scaling method. The standardization of z-scores is defined in Equation (1).

$$E' = \frac{E-\bar{M}}{\sigma_M} \qquad (1)$$

Where E' and E represent the new and original values for each data entry, respectively $\bar{M}$ denotes the mean, and σ_M is the standard deviation.

D. Feature Selection

Feature selection is a strategy for increasing machine learning model performance, lowering computing cost, and making model interpretation easier by identifying and selecting the most relevant input variables (features) from the dataset. Models that undergo feature extraction are more accurate, efficient, and resistant to overfitting because they are created from a smaller, more concentrated subset of data that is free of unnecessary, redundant, or noisy features. The goal of this method is to improve the efficiency of the models, reduce the amount of computation, and boost the level of model interpretability by picking the most informative features and deleting the unnecessary and redundant information.

E. Data Balancing using SMOTE

Data Balancing The process of balancing an uneven dataset in which one or two classes have significantly fewer samples than the rest, the class imbalance by interpolating between feature values between the minority sample and the closest within-class data, SMOTE creates extra data points in a minority class. By altering the class distribution, SMOTE is widely used as a benchmark for oversampling. It creates fresh data points for the minority group that are synthetic rather than merely

duplicates. As a result, the machine learning model's performance is enhanced and overfitting is reduced.

F. Data Splitting

A stratified split with a ratio of 80:20 was used to divide the train and test sets. The original dataset's proportionate class distribution was maintained in the training and testing sets by the use of a stratified split.

G. Proposed Convolutional Neural Networks (CNN) Model

Predictive threat modeming implements a deep learning strategy known as CNN. While learning features, CNNs use a multi-layered architecture that incorporates non-linear transformations. The visible layer shows the input data, which is a multidimensional data array known as a tensor. Grid-like data structures, such time-series data (a 1D grid sampling at regular intervals), 2D image pixels, 3D video frames, etc., frequently exhibit this architecture. This is followed by the extraction of many abstract features through a sequence of hidden layers. One example is the 2D convolution, which is calculated using Equation (2) using a two-dimensional kernel h and a two-dimensional input x.

$$(x * h)i, j = x[i, j] * h[i, j] = \sum n \sum m x[n, m] \cdot h[i - n][j - m] \quad (2)$$

The dot product of their weights with a little region that they are related to in the input.

After convolution, a feature map is generated at the filter output using a point-wise nonlinearity g and a bias term. Equation (3) illustrates how the filters of the feature map \h^l, or the l-th feature map at a specific convolutional layer, are defined by the coefficients or weights W^l, the input x, and the bias bl.

$$h^l_{i,j} = g(W^l * x)_{ij} + b_l \quad (3)$$

The 2D convolution is described by Equation (1), and the activation function is denoted by g (·).

A rectifier activation function, as defined in Equation (4), is a common one in deep neural networks.

$$g(x) = x^+ = max(0, x) \quad (4)$$

H. Evaluation Metrics

A set of performance metrics was used to evaluate the proposed design's performance. In the beginning, created a confusion matrix to show the results of the classification, which highlighted the numbers of right and wrong predictions for each class. The most important values, including True Positives (TP), False Positives (FP), True Negatives (TN), and False Negatives (FN), were extracted from this matrix. These numbers were used to compute the key metrics such as REC, ACC, PRE and F1 as indicated below:

Accuracy: The fraction of the cases in the sample of instances (input samples) the trained model correctly predicted as a proportion of the total cases. The solution is provided as Equation (5)-

$$Accuracy = \frac{\text{TP+TN}}{\text{TP+Fp+TN+FN}} \quad (5)$$

Precision: Precision is the percentage of the number of positive cases that are accurately predicted by the model. Precision shows. Equation (6) expresses how well the classifier predicts the positive classes.

$$Precision = \frac{\text{TP}}{\text{TP+FP}} \quad (6)$$

Recall: This statistic is the proportion of the number of events that should have been positive of the total number events that were correctly predicted as positive. Its mathematical formula is Equation (7).

$$Recall = \frac{\text{TP}}{TP+FN} \quad (7)$$

F1 score: It allows for a fair evaluation of PRE and REC by representing the harmonic mean of these measurements. It spans [0, 1]. Mathematically, it is given as Equation (8)-

$$F1 - score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (8)$$

IV. Results and Discussion

This section outlines the experimental setup, demonstrates how the proposed model operates during training and testing, and emphasizes its computational efficiency and evaluation process. An Intel(R) Core-i7 11370H CPU running 64-bit Windows 10 with 16 GB of RAM was used for the study. After training on the CICIDS 2017 dataset, the suggested model was assessed using ACC, PRE, REC, and F1 as key performance metrics. Table II shows that when the suggested CNN model is used for predictive threat modeling in cloud environments using the CICIDS dataset 2017, it is very effective and robust in all evaluation criteria. The model proved to be able to classify the vast majority of cases with a wonderful level of ACC of 99.14%. It has an ACC of 98.69% and REC of 99.11% which is the good dependability of it in reducing false positives and capturing almost all the instances of actual threats respectively. Also, the F1-score of 98.90 percent shows a fine balance between the REC and the ACC, which proves the overall efficiency and high superiority of the CNN model in cloud-based predictive threat detection.

TABLE II. Classification results of the proposed model, Predictive threat modeling in cloud environments using CICIDS 2017 dataset

Matrix	Convolutional Neural Networks (CNN) Model
Accuracy	99.23
Precision	98.32
Recall	99.15
F1-score	98.35

Fig. 4 shows the ACC of the proposed model's training and validation on the CICIDS 2017 binary dataset over several epochs. The graph shows ACC (y-axis, ranging from 0.84 to 1.0) versus epochs (x-axis, 0 to 100). Both training and validation curves demonstrate stable performance around 0.98-0.99 ACC, with a notable validation ACC drop occurring near epoch 80 before recovery, implemented within Industrial IoT (IIoT) Networks.

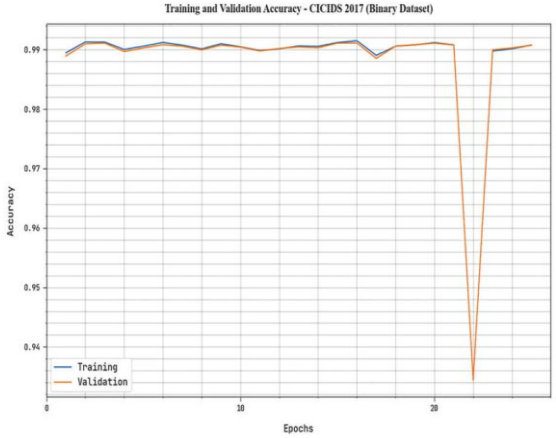

Fig. 4. Accuracy curve for the CNN Model

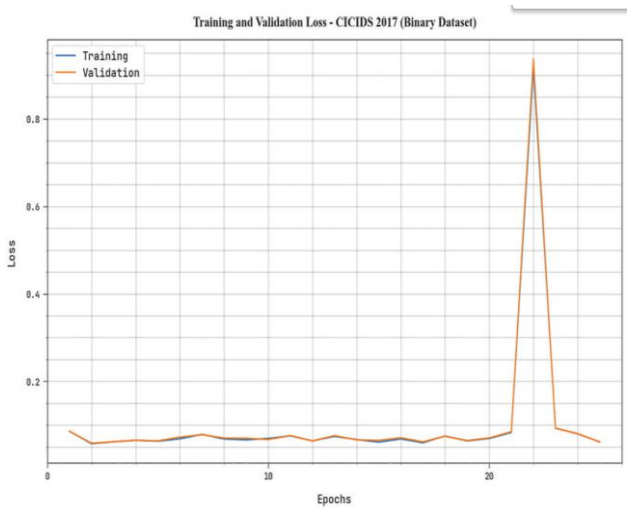

Fig. 5. Loss curve for the CNN Model

Fig. 5 displays the training and validation losses of the proposed CNN model across the epochs on the CICIDS 2017 binary dataset. The graph shows the value of the loss (y-axis 0-1.0) versus training iterations (x-axis). The two curves have well-steady loss of about 0.1 up to epoch 80 where a series of high spikes take place after which the model converges quickly, indicating efficient optimization of the model used in distributed IoT machine learning systems.

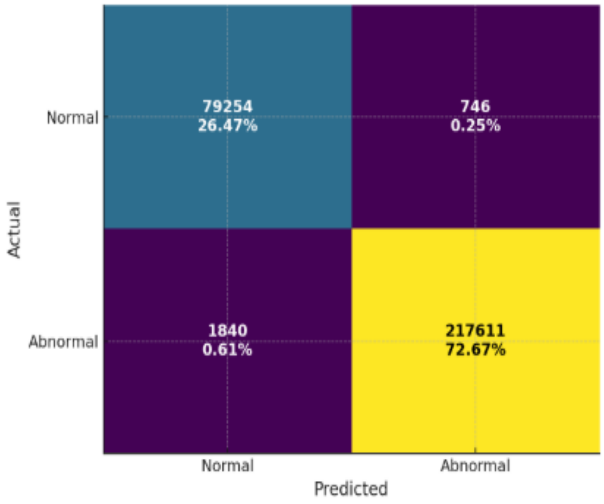

Fig. 6. Confusion Matrix for the CNN Model

The capacity of the suggested model to distinguish between normal and abnormal instances on the 2017 CICIDS data is shown by the confusion matrix in Fig. 6. Among the normal samples, 79, 254 samples (26.47) were classified correctly and only 746 samples (0.25) were misclassified as abnormal. In the case of abnormal traffic, the model correctly identified 217,611 samples (72.67%), and only 1,840 samples (0.61) were wrongly classified as normal. These findings indicate that the model is very accurate and effective especially in detection of abnormal traffic.

A. Comparative Analysis

Table III compares the proposed CNN model's ACC to other existing models in order to measure its usefulness. Using the CICIDS 2017 dataset to train various machine learning models on cloud-based predictive threat modelling, it was found that these models varied significantly in all four-assessment metrics: ACC, PRE, REC, and F1. With an F1 of 83.45, ACC of 84.1%, PRE of 86.04%, and REC of 84.13%, LR performed the worst. Although REC and PRE statistics were not provided, the LSTM model achieved an ACC of 93.78% and an F1 of 90.9%. With 97.7% ACC, 98% PRE, 90% REC, and 94% F1, the MLP model produced superior results. The CNN model performed the best in general, with the ACC being 99.23%, PRE is 98.32%, REC is 99.15% and F1 is 98.35%, which means that it is the best at predictive threat modeling in a cloud environment.

TABLE III. COMPARISON OF DIFFERENT MACHINE LEARNING MODELS FOR PREDICTIVE THREAT MODELING IN CLOUD ENVIRONMENTS ON CICIDS2017 DATASET

Model	Accuracy	Precision	Recall	F1-score
LR[31]	84.13	86.04	84.13	83.45
LSTM[32]	93.78	-	-	90.95
MLP[33]	97.7	98	90	94
CNN	99.23	98.32	99.15	98.35

The suggested CNN model with an outstanding ACC of 99.23% on the CICIDS 2017 dataset is highly beneficial as it allows a very high degree of reliability in predictive threat modeming in cloud settings. Its high performance guarantees effective involvement of normal and abnormal traffic patterns and the reduction of the risk of misclassifications that lead to higher network security. The CNN is more efficient and scalable, unlike the traditional machine learning methods, which require a lot of manual engineering to extract complex feature. The model has shown high efficiency in computation, real-time processing, and remarkably low false positive rates (0.25%), which make it the best in resource-constrained industrial IoT applications, as well as in enabling flawless deployment on the cloud and adjusting the threats in real-time.

V. CONCLUSION AND FUTURE WORK

The IIoT connects industrial devices, sensors, and equipment with cloud-based platforms to enable data-driven decision-making, automation, and real-time monitoring. Cyber threats pose a significant risk to IIoT systems due to the large volume and diverse nature of network traffic. Conventional security solutions, including signature-based intrusion detection. As a result, machine learning-based predictive threat modeling has become an effective technology that can improve the safety of IIoT networks on the cloud. This paper suggests a DL-powered threat detection system in the network traffic through a CNN and using the CICIDS 2017 dataset. Maximum preprocessing, such as data cleaning, normalization, feature selection, and data balancing with SMOTE was implemented to improve model performance. CNN model was created to automatically learn and extract multidimensional traffic complex spatial-temporal patterns to provide correct classification between benign and malicious traffic. The experimental findings prove that the proposed CNN model is more accurate, which was 99.23%, than the traditional ML models, such as LR (84.1%), LSTM (93.78%), and MLP (97.7%). These results indicate the high level of CNNs to identify complex patterns in the IIoT network traffic and offer an efficient remedy in detecting threats proactively in cloud systems. The suggested framework is a solid methodology of preventing IIoT networks against the new cyber threats. The next steps in the work will be taking an interest in transformer-based and graph neural network models and testing the framework on the actual IIoT cloud implementation to guarantee greater applicability and resistance to changing attacks.

REFERENCES

[1] Abhishek and P. Khare, “Cloud Security Challenges: Implementing Best Practices for Secure SaaS Application

Development," Int. J. Curr. Eng. Technol., vol. 11, no. 06, pp. 669–676, Nov. 2021, doi: 10.14741/ijcet/v.11.6.11.

[2] K. Shukla, N. Patel, and H. Mistry, "Securing the Cloud: Strategies and Innovations in Network Security for Modern Computing Environments," Int. Res. J. Eng. Technol., vol. 11, no. 04, 2024.

[3] C. Hazman, A. Guezzaz, S. Benkirane, and M. Azrour, "A smart model integrating LSTM and XGBoost for improving IoT-enabled smart cities security," Cluster Comput., vol. 28, no. 1, p. 70, 2024, doi: 10.1007/s10586-024-04780-1.

[4] S. R. Alotaibi et al., "Explainable artificial intelligence in web phishing classification on secure IoT with cloud-based cyber-physical systems," Alexandria Eng. J., vol. 110, pp. 490–505, 2025, doi: https://doi.org/10.1016/j.aej.2024.09.115.

[5] S. Gupta, "Hybrid Cloud Integration and Multicloud Deployments A Comprehensive Review of Strategies, Challenges, and Best Practices," Int. J. Adv. Res. Comput. Sci., vol. 16, no. 2, pp. 59–64, Apr. 2025, doi: 10.26483/ijarcs.v16i2.7233.

[6] Z. A. Abdulkader, "Cloud data security mechanism using the lightweight cryptography," 2022. doi: 10.1016/j.ijleo.2022.170084.

[7] M. S. Christo, V. E. Jesi, U. Priyadarsini, V. Anbarasu, H. Venugopal, and M. Karuppiah, "Ensuring Improved Security in Medical Data Using ECC and Blockchain Technology with Edge Devices," Secur. Commun. Networks, vol. 2021, pp. 1–13, Oct. 2021, doi: 10.1155/2021/6966206.

[8] S. B. Shah, B. Boddu, N. Prajapati, and S. A. Pahune, "AI-Powered Advanced Intrusion Detection for Securing Cloud Environments Against Network Attacks," in 2025 Global Conference in Emerging Technology (GINOTECH), IEEE, May 2025, pp. 1–7. doi: 10.1109/GINOTECH63460.2025.11076673.

[9] S. G. Jubin Thomas, Kirti Vinod Vedi, "The Effect and Challenges of the Internet of Things (IoT) on the Management of Supply Chains," vol. 8, no. 3, pp. 874–878, 2021.

[10] H. S. Chandu, "Enhancing Manufacturing Efficiency: Predictive Maintenance Models Utilizing IoT Sensor Data," IJSART, vol. 10, no. 9, 2024.

[11] H. S. Chandu, "A Review of IoT-Based Home Security Solutions: Focusing on Arduino Applications," TIJER – Int. Res. J., vol. 11, no. 10, pp. a391–a396, 2024.

[12] K. M. R. Seetharaman, "Internet of Things (IoT) Applications in SAP: A Survey of Trends, Challenges, and Opportunities," Int. J. Adv. Res. Sci. Commun. Technol., vol. 3, no. 2, pp. 499–508, Mar. 2021, doi: 10.48175/IJARSCT-6268B.

[13] S. Thangavel, K. Narukulla, and R. Sundaram, "Edge-Enabled Distributed Computing for Low-Latency IoT Applications: Architectures, Challenges, and Future Directions," Int. J. Emerg. Res. Eng. Technol., vol. 3, no. 1, pp. 28–41, 2022, doi: 10.63282/3050-922x.ijeret-v3i1p104.

[14] A. Derhab et al., "Blockchain and Random Subspace Learning-Based IDS for SDN-Enabled Industrial IoT Security," Sensors, vol. 19, no. 14, p. 3119, Jul. 2019, doi: 10.3390/s19143119.

[15] R. Patel, "Optimizing Communication Protocols in Industrial IoT Edge Networks: A Review of State-of-the-Art Techniques," Int. J. Adv. Res. Sci. Commun. Technol., vol. 4, no. 19, pp. 1–12, 2023.

[16] R. Patel, "Artificial Intelligence-Powered Optimization of Industrial IoT Networks Using Python-Based Machine Learning," J. Eng. Technol. Adv., vol. 3, no. 4, pp. 138–148, 2023, doi: 10.56472/25832646/JETA-V3I8P116.

[17] S. Narang and A. Gogineni, "Zero-Trust Security in Intrusion Detection Networks: An AI-Powered Threat Detection in Cloud Environment," Int. J. Sci. Res. Mod. Technol., vol. 4, no. 5, pp. 60–70, Jun. 2025, doi: 10.38124/ijsrmt.v4i5.542.

[18] S. K. Senthilkumar Thangavel, Arnav Kotiyal, Ancy Thomas, Harshal Patil, "Robust Authentication Protocols for IoT Devices in High-Density Networks," 2024 Int. Conf. Distrib. Syst. Comput. Networks Cybersecurity, pp. 1–7, 2024.

[19] V. Shah, "Analyzing Traffic Behavior in IoT-Cloud Systems : A Review of Analytical Frameworks," Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol., vol. 9, no. 3, pp. 877–885, 2023.

[20] V. Shah, "Traffic Intelligence in IoT and Cloud Networks: Tools for Monitoring, Security, and Optimization," Int. J. Recent Technol. Sci. Manag., vol. 9, no. 5, 2024, doi: 10.10206/IJRTSM.2025894735.

[21] S. Al-Farsi, M. M. Rathore, and S. Bakiras, "Security of Blockchain-Based Supply Chain Management Systems: Challenges and Opportunities," Appl. Sci., vol. 11, no. 12, p. 5585, Jun. 2021, doi: 10.3390/app11125585.

[22] P. Piyush, A. A. Waoo, M. P. Singh, P. K. Pareek, S. Kamal, and S. V. Pandit, "Strategizing IoT Network Layer Security Through Advanced Intrusion Detection Systems and AI-Driven Threat Analysis," J. Intell. Syst. Internet Things, vol. 24, no. 2, pp. 195–207, 2024, doi: 10.54216/JISIoT.120215.

[23] B. Yadav, D. D. Rao, Y. Mandiga, N. S. Gill, P. Gulia, and P. K. Pareek, "Systematic Analysis of Threats, Machine Learning Solutions and Challenges for Securing IoT Environment," J. Cybersecurity Inf. Manag., vol. 14, no. 2, pp. 367–382, 2024.

[24] A. S. Haider, M. T. Nafis, I. R. Khan, S. T. Owais, and N. Fatima, "Hybrid GAN-Based Transfer Learning Model for Advanced Threat Detection in Cloud Computing," in 2025 3rd International Conference on Device Intelligence, Computing and Communication Technologies (DICCT), 2025, pp. 531–535. doi: 10.1109/DICCT64131.2025.10986358.

[25] R. K. Ch, S. Nimmala, I. Batra, and A. Malik, "A Comparative Study of Machine Learning Techniques for Threat Detection and Enhancing Security in Industrial IoT," 2025, pp. 295–310. doi: 10.4018/979-8-3693-6135-1.ch011.

[26] E. E. Mohammed, R. Y. S. Naji, A. A. Hussein, M. A. Saeed, and R. A. M. Al Selwi, "Anomaly Detection System For Secure Cloud Computing Environment Using Machine Learning," in 2025 5th International Conference on Emerging Smart Technologies and Applications (eSmarTA), 2025, pp. 1–9. doi:

10.1109/eSmarTA66764.2025.11132254.

[27] A. Houkan et al., "Enhancing Security in Industrial IoT Networks: Machine Learning Solutions for Feature Selection and Reduction," IEEE Access, vol. 12, no. September, pp. 160864–160883, 2024, doi: 10.1109/ACCESS.2024.3481459.

[28] A. Vinolia, N. Kanya, and V. N. Rajavarman, "Machine Learning and Deep Learning based Intrusion Detection in Cloud Environment: A Review," in Proceedings - 5th International Conference on Smart Systems and Inventive Technology, ICSSIT 2023, 2023. doi: 10.1109/ICSSIT55814.2023.10060868.

[29] M. S. Hossain and M. S. Islam, "Economic Denial of Sustainability Attack Detection Using Machine Learning," in 2023 26th International Conference on Computer and Information Technology (ICCIT), 2023, pp. 1–6. doi: 10.1109/ICCIT60459.2023.10441045.

[30] G. Tiwari and R. Jain, "Detecting and Classifying Incoming Traffic in a Secure Cloud Computing Environment Using Machine Learning and Deep Learning System," in 2022 IEEE 7th International Conference on Smart Cloud (SmartCloud), IEEE, Oct. 2022, pp. 16–21. doi: 10.1109/SmartCloud55982.2022.00010.

[31] S. Farhat, M. Abdelkader, A. Meddeb-Makhlouf, and F. Zarai, "Evaluation of DoS/DDoS Attack Detection with ML Techniques on CIC-IDS2017 Dataset," in International Conference on Information Systems Security and Privacy, 2023. doi: 10.5220/0011605700003405.

[32] A. Parashar, M. Sahare, and A. Shrivastava, "Detection of Cyber-Attacks in Cloud Computing using Cascaded Machine Learning Algorithm," vol. 11, no. 3, pp. 234–239, 2024.

[33] Z. Xu and Y. Liu, "Robust Anomaly Detection in Network Traffic: Evaluating Machine Learning Models on CICIDS2017," arXiv, 2025.

