National College of Ireland

Real-Time Detection of IoT Cyber Threats Using Advanced Deep Learning Methods

MSc Research Project
MSc in Data Analytics

Tushar Gharpure
Student ID: x23289902

School of Computing
National College of Ireland

Supervisor: Shubham Subhnil

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Tushar Gharpure
Student ID:	x23289902
Programme:	MSc in Data Analytics
Year:	2025
Module:	MSc Research Project
Supervisor:	Shubham Subhnil
Submission Due Date:	11/08/2025
Project Title:	Real-Time Detection of IoT Cyber Threats Using Advanced Deep Learning Methods
Word Count:	8773
Page Count:	27

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Tushar Gharpure
Date:	9th August 2025

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Real-Time Detection of IoT Cyber Threats Using Advanced Deep Learning Methods

Tushar Gharpure
x23289902

Abstract

The growth of the Internet of Things (IoT) has disrupted modern industry but has also exposed networks to an expanding range of cyber threats. Resource-constrained IoT devices, often deployed in unsecured and heteronomous systems, are inherently vulnerable to exploitation via Distributed Denial of Service (DDoS), brute-force intrusion, and malicious scanning. Conventional intrusion detection systems (IDS) lack sufficient flexibility or processing power to address these vulnerabilities by relying on predefined signatures of malicious activity and requiring a reengineering process to adapt to new and evolving issues. In this study, we proposed a deep learning (DL) based intrusion detection framework for IoT security using three architectures (Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM), and a hybrid Convolutional LSTM (Conv-LSTM) model). Experimental results demonstrated that both CNN and LSTM offered poor or decreased performance when compared to the Conv-LSTM model. When evaluating the hybrid Conv-LSTM model under the experimental framework based upon the benchmark IoT dataset, the Conv-LSTM model accomplished near-perfect accuracy, precision, recall, and F1-score, and significantly lower false positive and false negative detections. The proposed framework was also deployed as a real-time web application capable not only of live traffic monitoring but also of providing immediate notification of threats in real-time. Overall, my findings support the claim that the Conv-LSTM model architecture provides a high-performance, scalable, and beyond a doubt accurate approach to next-generation IoT intrusion detection.

1 Introduction

The overview of the “Internet of Things” (IoT) introduces an emerging paradigm of communication and engagement of users through sensors, which eliminates direct involvement. The purpose of IoT has become a promising solution in both industrial aspects and for individual users, where data analytics performs a crucial role based on real-time data processing. Data analytics which, when performed through IoT, identifies the need for developing novel methodologies that can work with a limited budget of computation. When focusing on data analysis, the detection of an unusual state in the system is mandatory, which is known as anomaly detection or outlier detection. In the IoT system, this anomaly detection demonstrates certain checkpoints determining incoming traffic across various stages. Thus, the process is considered to be significant in data cleaning and classification.

The detection of IoT anomalies is a recognisable discussion in the literature, which draws significance in terms of cybersecurity and safe handling of the computing system. The relevance of IoT anomaly detection identifies a necessary approach to classify and analyse unusual IoT data patterns, which is effective in providing actionable information from multiple sectors. However, with a rising data figure due to significant reliance on the IoT system in recent times, a dilemma has been observed in effective anomaly detection using traditional methods. Therefore, modern methods using machine learning and deep learning algorithms are implemented in practice to ensure reliability, efficiency, as well as security across connected devices in the IoT world. Evidence from the literature and analytical reports suggests that the detection of anomalies identifies potential issues beforehand, to prevent escalation, thus offering businesses necessary and valuable insights.

Upon understanding the priority, this study will investigate the purpose of real-time anomaly detection in the IoT system using advanced deep learning algorithms. Deep learning models are advanced machine learning algorithms with highly effective features, including learning patterns as well as a relational approach to the dataset, which are robust in IoT anomaly detection. This research proposed a deep learning based IoT intrusion detection framework using Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM) networks, and a combined Conv-LSTM. The Conv-LSTM model was used to model the spatial and temporal dependencies present in IoT network traffic, bringing together the CNN feature extraction capacity and the sequential modeling of the LSTM. The results were evaluated using a benchmark IoT dataset and demonstrated superior performance on numerous metrics and the model near-perfect detection accuracy. Further, we developed a real time web application that leveraged the best performing model for continuous monitoring and providing instant alerts for road friction threats. This research paves the way for the next generation IDS solutions that are robust, scalable and capable of addressing the ever-evolving threats for IoT environments.

1.1 Research Objectives

This research aims to enhance IoT security by overcoming the limitations of traditional intrusion detection systems through advanced deep learning approaches. The specific objectives are:

- To identify the limitations of traditional intrusion detection systems in adapting to evolving IoT threats and detecting zero-day attacks.
- To design and implement three deep learning architectures: Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM), and a hybrid Convolutional Long Short-Term Memory (Conv-LSTM) for multi-class IoT attack detection.
- To compare the performance of these models using evaluation metrics such as accuracy, precision, recall, and F1-score to determine the most effective architecture.
- To develop and deploy a real-time web-based intrusion detection application integrating the best-performing model for live traffic monitoring and instant threat alerts.

1.2 Research Question

Based on the identified gaps and objectives of this study, the central research question guiding this work is:

- How can an intrusion detection system for IoT networks be designed using deep learning techniques to effectively detect and classify diverse cyber-attacks in real time with high accuracy and minimal false alarms?

2 Related Work

This section reviews the literature related to IoT intrusion detection and anomaly detection techniques and describes the current techniques, their advantages, disadvantages, challenges of effectively detecting threats in real-time.

2.1 Real-time anomaly detection in the IoT system - Classification and Challenges

The anomaly detection in the IoT system is widely discussed in the literature, where some applications are central to the network security vulnerabilities, while some studies are mainly concerned about the growth potential of the anomaly detection process. According to the information provided by Chatterjee and Ahmed (2022), anomaly detection in IoT is categorised as rare events, which show a deviation in a data point from its original behaviour/pattern. In binary classification of anomalies, the model approximation, based on its suitability as per the data behaviour, has been considered crucial. At the same time, Chatterjee and Ahmed (2022) explained that situational complexities require a specific detection strategy, which is ideal for each anomaly detection application. The evidence provided by Yang and Zhang (2023), IoT anomaly detection holds paramount importance due to the extensive sensor data utilisation by industries like education, finance, agriculture and retail. As mentioned in the above study, seamless device integration increases substantial data use, which further exposes organisations to cybersecurity vulnerabilities. Thus, reliability and system integrity are mandatory to enhance the overall performance.

Over the past few decades, IoT has attracted a significant number of users, especially in business organisations across multiple industries. As stated by Sgueglia et al. (2022) the rapid development of the IoT system has prompted both individuals and industries to adopt the highly integrated network paradigm, thus making the automation process more feasible. While the data collection as well as the monitoring process are continuously improving, Sgueglia et al. (2022) in their study mentioned that several issues are persistent, which are further categorised into standard issues and anomalies. Notably, standard issues are focused on energy consumption, heterogeneity in devices and standardisation. On the other hand, IoT anomalies are highly concerning aspects which impact privacy, security, and system vulnerability. According to the information provided by Gaddam et al. (2020) IoT sensors deployment within a harsh environment leads to prediction malfunctioning with incorrect sensor-based operations. Besides, IoT sensors are considered to be an inexpensive component, which is prone to malfunctioning.

A deeper knowledge of IoT sensor malfunctioning identifies the gathering of corrupted data and erroneous readings, which thereby impact the accuracy and reliability rate. Hence, it can be stated that IoT sensor malfunctioning, when operating with high data volumes, impacts the operational accuracy, which therefore requires precise monitoring of the behaviour as well as performance. The monitoring process, as explained by Gaddam et al. (2020) is actually the outlier (anomaly) detection process in sensors, which influences the overall data quality. While this is true, research evidence presented by Nizam et al. (2022) shows that multiple connected devices & smart sensors in the IoT are operating in a highly dynamic and heterogeneous environment. The time-stamped data is considered the key component for IoT automation, while it holds the propensity to influence industrial processes. Considerably, the data pose significant challenges in effective anomaly detection, thus requiring significant improvement. Over the years, evolution in the anomaly detection process has been identified to gain more accurate and reliable outcomes.

Distinctly, real-time anomaly detection in IoT devices based on streaming sensor data is a key priority to understand and classify rare events as outliers. As such, the dataset has gained priority in structuring the detection process using different classification models and anomaly detection methods. Cook et al. (2020) explained that the anomaly detection process is specific and focused on users' use cases, thus requiring expert knowledge regarding the dataset, method and the situation. As such, gathering evidence-based information from the literature has illustrated both opportunities and drawbacks of methods in assessing sensor data in real-time IoT anomaly detection, when encountered in applications. Herein, it is indeed imperative to consider that managing as well as analysing rare events within extensive sensor data is stimulated through expert knowledge on methods in adopting the detection techniques. At the same time, it is also acknowledged that analysing time series data, which is the most common process of IoT anomaly detection, needs specific consideration of both time resources in a constrained situation. This addresses dimensionality issues, enhances real-time monitoring and effective localisation of IoT anomalies.

2.2 IoT anomaly detection using Time-Series Methods

In recent years, one of the most important purposes served by the IoT system is smart infrastructure for urban landscapes. According to the information provided by Muntean (2024), smart cities apply advanced technologies and data analytics as well as interconnected systems in developing efficient and sustainable environments. However, as already explained above, the generation of extensive sensor data increases IoT system vulnerabilities with rising anomalies/outliers in the generated data. Thus, detecting the data traffic using potential methods has become a mandatory paradigm in research. According to the information provided by Priyadarshini et al. (2022) the detection of anomalous behaviour of IoT data determines whether the system is trustworthy, thus applying numerous methods for analysis and classification of this data. As such, evidence-based understanding of this detection process has identified the purpose of time-series methods, for example, the "AutoRegressive Integrated Moving Average" (ARIMA), which is further compared with other methods. Bestowing focus on the experimental observation, it has been identified that when compared with other methods, ARIMA has outperformed by providing a reliable outcome in terms of RSM, MSE & MAE values.

Another study presented by Giannoni et al. (2018) introduced the relevance of in-situ sensors as well as “Wireless Sensor Networks” (WSN) in the recent decade due to their applications in multiple fields. However, it has been identified that with extensive data generation and dependence, these sensors are prone to providing erroneous readings and data corruption at the time of transmission. This issue, therefore, creates a problem in ensuring data quality. When exploring the concern, this study indeed introduces essential facts regarding time series data and the necessity of automated detection of anomalies in this data. However, the study has provided limited evidence on methods that are traditionally used by experts in detecting erroneous readings and anomalous behaviour of sensor data. From the above information, it is understandable that anomaly detection has become an important part of “time-series analysis”, for example, prediction as well as forecasting based on this data. Considering the focus, the information provided by Apostol et al. (2021) explained that abnormal behaviour detection in IoT sensors demands an efficient and reliable detection system, where machine learning models have outsmarted traditional models. This identified information, therefore, suggested that improvement in the detection process is mandatory to ensure the accuracy and reliability of sensor data.

2.3 IoT anomaly detection using Machine Learning Models

Apart from the above knowledge integration on traditional methods and approaches to anomaly detection in the IoT system, for the past few decades, research attention has been given to improved detection algorithms, such as machine learning models. According to the information provided by Hasan et al. (2019) machine learning (ML) applications have become a key research focus over the past few decades, where the data-driven infrastructure that individuals are accustomed to in recent times has become a potential contributor to the progress. Amid this understanding, the above study has provided specific evidence on malicious attacks and anomalies in the IoT infrastructure, which is indeed a concern in recent times. Some growing attacks in terms of cyber threats include DOS, malicious probing and spying. Besides, anomalies, as already explored in this research study, are erroneous readings and abnormal behaviour of IoT sensor data Diro et al. (2021). Thus, predicting these anomalies has become a key focus as part of machine learning applications. Hasan et al. (2019) in their study introduce various ML models, including “Logistics Regression” (LR), “Support Vector Machine” (SVM), “Decision Tree” (DT), “Random Forest” (RF) and “Artificial Neural Network” (ANN).

The evaluation of the efficiency of these models has been performed using various metrics, which compare each model in terms of accuracy, precision rate, F1-score and recall rate. As per the experimental result, three ML models - ANN, DT and RF have outperformed other models through an estimated performance of 99.4%. Arguably, another study introduced by Mukherjee et al. (2023) has explained how IoT abnormalities, especially in sensor data, are increasingly detected using ML models, while comparing their efficiency and reliability rates, in terms of the “state of the art” condition. In this study, it has been identified that the prediction of anomalies is performed by training and testing each model using time series data, which provides an accuracy of 99.4% in the first time, while 99.99% accuracy in the later period. With the rapid evolution of growing technologies, security threats are a prominent research focus that is repeatedly

discussed and emphasised by practitioners and researchers. As such, Haji and Ameen (2021), explained that the implementation of ML technology is a powerful approach in detecting suspected threats or anomalies in the IoT system. With the introduction of advanced detection methods like machine learning, deep learning and hybrid techniques, real-time detection of erroneous or abnormal data behaviour in IoT is feasible and reliable.

In line with the above demonstration, Sahu and Mukherjee (2020), provide relevant insights into the anomaly detection process in IoT using machine learning algorithms. The study introduces two models - logistic regression (LR) and artificial neural network (ANN), both of which are trained using a 3.5 lakh dataset. The prediction accuracy has been determined for each model, which indicates that LR has outperformed the ANN model by 99.99% accuracy, indicating that the state-of-the-art algorithm is effective in identifying network threats and data anomalies in the IoT system.

2.4 Detection of IoT anomaly using Deep Learning Models

It is indeed appreciable that machine learning models are a paradigm of importance in anomaly prediction in the IoT system. However, given the condition of extensive data generation in recent years, a concern regarding the detection reliability has been identified. Hence, experts have introduced more improved models such as deep learning algorithms and neural architectures. According to the information provided by Ullah and Mahmoud (2021), a growing dependence on IoT devices identifies its relevance in terms of both opportunities and challenges. As such, the study has prioritised anomaly detection in IoT using deep learning models, while indicating the inefficiency of traditional ML methods in contemporary prediction of anomalous behaviour of sensor data. The study emphasises the significance of a deep neural architecture, the CNN model in various dimensional configurations (D, 2D and 3D). Upon using the MQTT-IoT-IDS2020 dataset, besides IoT datasets, the model has been trained. Further, the transfer learning (TL) algorithm has been used to implement both binary & multi-class classification through the CNN architecture. Indicating the experimental data, Ullah and Mahmoud (2021), informed that the model has shown its relevance with higher accuracy in IoT anomaly detection, which is further compared to various other deep models.

Indicating the above information, it is understandable that deep learning models, precisely the convolutional neural network (CNN), have gained significant attention due to their classification and detection efficiency for complex, abnormal data. Contrary to this classification, Ahmad et al. (2021) explained the importance of other deep neural models such as “Gated Recurrent Unit” (GRU), “Long Short-Term Memory” (LSTM), and “Recurrent Neural Network” (RNN). It is specified in this study that IoT has been continuously revolutionising with an expansion in its applications across multiple fields. As such, the study equally explored IoT network security vulnerabilities under erroneous reading and abnormal data behaviour. Upon reviewing the concern, the literature emphasises and compares the performance of different deep learning models by training and testing each model using the IoT-Botnet-2020 dataset. As per the experimental observation, an improvement in the detection accuracy of each model is improved by 0.57-2.60%. At the same time, it has been observed that the “false-alarm rate”, which is a common issue in traditional ML models, has been reduced by 0.23-7.98%. Focusing on the experimental result, an interesting fact has been noted that the overall detection accuracy of

all DL models included in the above study has provided an accuracy rate of 99%, thus demarcated as the most efficient and reliable anomaly detection models at present.

A considerable focus on real-time anomaly detection in the IoT system has achieved remarkable attention in research. The literature emphasis regarding the identification of abnormal behaviour of IoT sensor data has become increasingly challenging due to the enormous data generation with continuous dependency Sharma et al. (2019). Herein, it is notable that analysing normal & anomalous behaviour of data has identified the key implication of deep learning models to address anomaly, despite the complications. However, the focus is still evolving, which therefore provided a scope from this literature evidence to further understand the contribution of deep learning models in real-time anomaly detection in the IoT system.

2.5 Identified Gaps in Literature Review

The literature review indicates three major gaps in the field regarding IoT intrusion detection. Many traditional and machine learning approaches face issues adapting to new IoT threats. Traditional approaches rely on static signatures, whereas machine learning approaches do not utilise extensive features, which limits their adaptability to threats such as zero-day attacks. In addition, many prior studies only include a binary classification of network traffic (normal versus attack), overlooking the complexity of detecting multiple classes of IoT attacks. Also of concern are the many difficulties associated with real-time deployment, such as a high rate of false positives and difficulties in handling large-scale streaming data. Therefore, we propose a new deep learning-based framework that uses a convolution neural network (CNN), long short-term memory (LSTM) and a novel architecture that combines both modules into a hybrid Convolution LSTM architecture for robust multi-class IoT attack detection. The best deep learning model is deployed into a real-time web-based system capable of monitoring live traffic and providing immediate threat alerts.

3 Methodology

The methodology section of this research outlines the systematic way in which the IoT intrusion detection system was developed and evaluated. An overview of the process is provided in Figure 1, which is a high-level view of the order of steps involved in the study. The detailed overview of each step is discussed in further subsections.

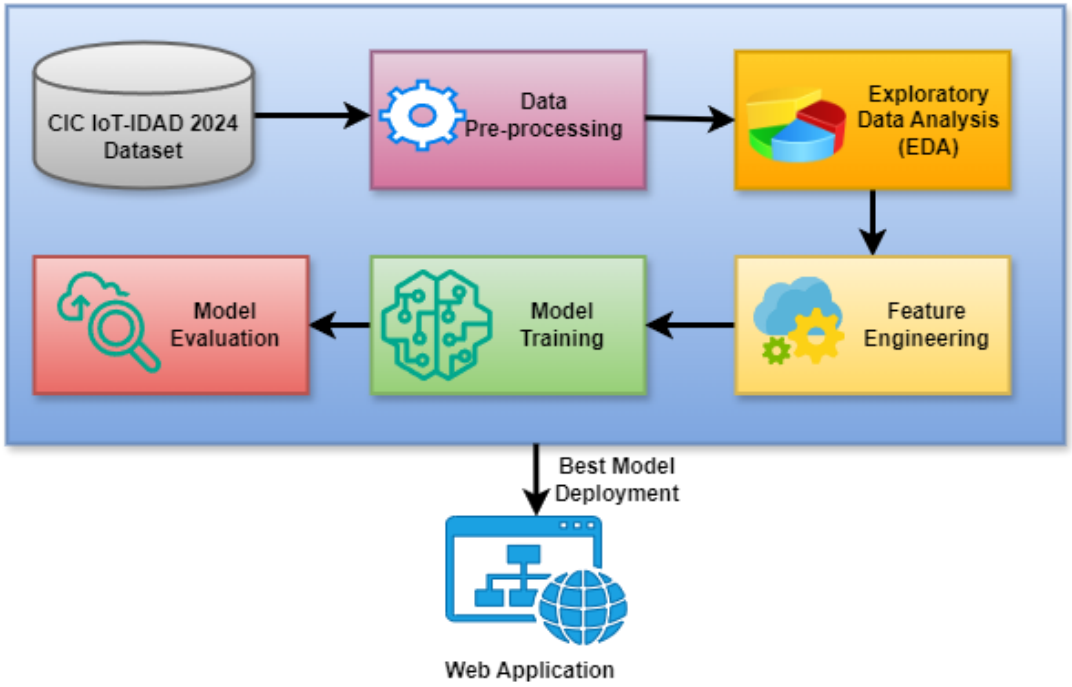

Figure 1: Methodology workflow for IoT intrusion detection

3.1 Dataset Description

The IoT-IDAD 2024 dataset is a newly developed, high-quality benchmark dataset that was designed to further research in intrusion detection systems (IDS), which is designed specifically for IoT networks Rabbani et al. (2024). The dataset was purposefully designed with modern threat scenarios in mind, and it captures benign and malicious network traffic from a variety of realistic IoT attack simulations. It contains many different types of attacks, which include: Denial of Service (DoS), Distributed Denial of Service (DDoS), Brute Force, Web-based attacks (XSS, SQL Injection, Uploading), and Network Scanning. We obtained the dataset from the official CIC website and used both the packet and flow-level CSV files for each attack campaign. The full dataset contained attacks of numerous types, but we decided to narrow our analysis to ten different classes involving Benign, DoS, DDoS (ACK and HTTP Floods), BruteForce (Dictionary), Web-based (3 types), and Scan. For each of the classes, each packet and flow-based data hass been merged based on the packet and flow identifiers (i.e., source/destination IPs and ports), allowing us to see a single view of the network behavior.

3.2 Data Preprocessing

After downloading the dataset, We thoroughly cleaned the column names by making them lowercase and removing unnecessary white spaces so it would all be consistent. We then merged the packet received data and flow data together. This was done on common column data (like IP addresses and ports), which allowed us to pull all of the data into a single data set for each attack type. When we finished the merging step, we named each data set with that respective attack type and combined everything into one overall dataset. In the final step, we dropped all duplicate rows so there would be no repeated

information. We continued by checking for missing values. All columns that had too many missing rows (greater than 80%), were dropped, and where possible, we filled in the missing data by averaging. We found that some columns had the same data (like 'src_ip' and 'src ip'), so we dropped one of the duplicates. After going through all of the cleaning steps, we ended up with a final cleaned and consistent dataset that was ready for machine learning processes. The data set was made up of 125,519 rows and 203 useful features.

3.3 Exploratory Data Analysis

To gain a deeper understanding of the dataset and identify patterns that could influence model performance, we conducted an exploratory data analysis (EDA). We performed EDA to understand data patterns, detect anomalies, and identify class imbalance in attack types.

The bar chart in Figure 2 below depicts the frequency of the attack categories found within the collection of records, measuring the data in terms of the number of records in that particular attack type. We see that 'Scan' attacks are very frequent with 57,974 counts, followed by 'Benign' (24,754) and 'DDoS' (23,979), followed by 'DoS' (10,602), 'Web-based' (4,705) and 'BruteForce' (3,505), which appear less frequently. This distribution highlights the aspects related to class imbalance, that certain types of attack arise with greater frequency than others.

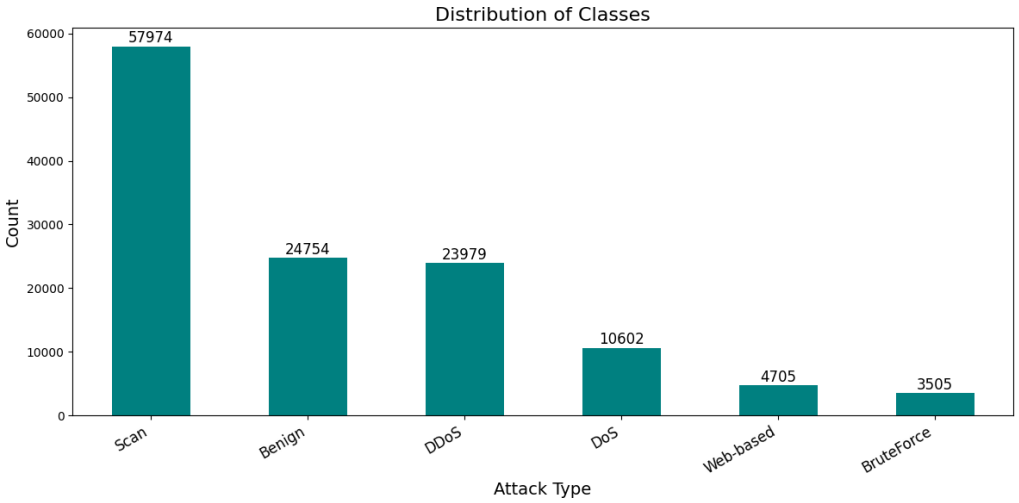

Figure 2: Distribution of Attack Types

With another analysis, the pie chart in Figure 3 depicts the percentage of all network protocols represented in the dataset. TCP is the leading transport protocol appearing in the data at (77.7%), followed by UDP (22.2%), with a small percentage classified as Other (0.1%). The predominant use of TCP suggests that the network communication captured with the attacks was primarily connection-oriented, while a connectionless UDP transport protocol was still used in a significant percentage and may be linked to specific types of attacks, such as DDoS floods.

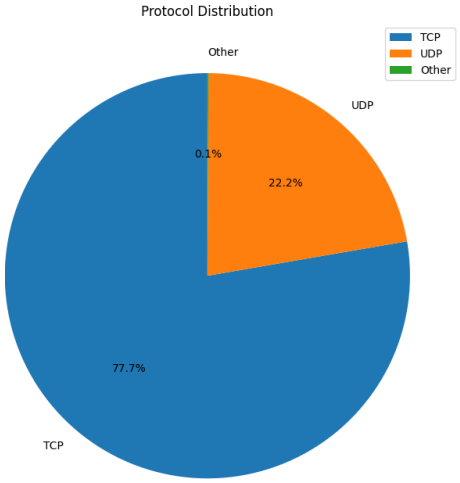

Figure 3: Protocol Usage Distribution in the dataset

The box plot in Figure 4 depicts the differences in average packet sizes between different types of attack and normal traffic. DoS attacks stand out as having the greatest median packet size, larger than all other attack types, and exhibiting the widest spread of packets (due to maximum packets being up to 1.500), indicating the loading of heavy bandwidth intended to overwhelm targets. DDoS attacks also demonstrate larger average packet sizes, but there is slightly more variance in sizes compared to DoS attacks. In comparison, Scan and Web-based attacks demonstrate the presence predominantly of small packets, consistent with rapid probes for scanning or lightweight requests for browsing. Benign traffic contains a large spread of packet sizes as well but mostly clusters around small packet sizes. However, BruteForce attacks limited themselves to very small packets.

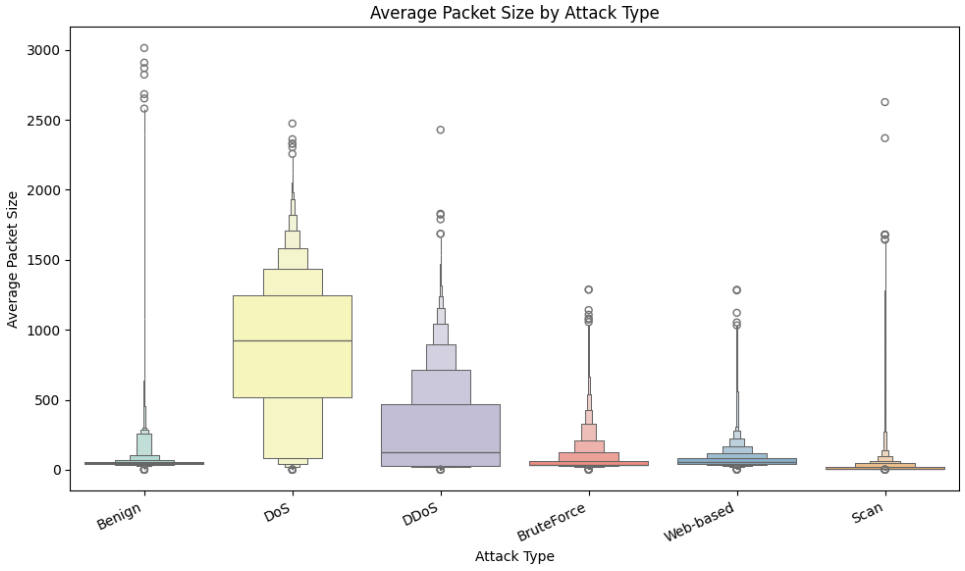

Figure 4: Average Packet Size Variation Across Attack Types

The grouped bar graph in Figure 5 indicates the number of attacks for each type across the five largest device manufacturers. Raspberry Pi Trading Ltd stood out with

an enormous number of Scan attacks (45,020), though large amounts of these attacks were a combination of DDoS and DoS attacks, leading to a conclusion of frequent targeting or misuse of Raspberry Pi's in scanning campaigns. The Fibar Group shows mostly benign traffic overall, but still exhibits traces of multiple attack types. Luxshare Precision Industry and Amcrest Technologies show a more normalized distribution of multiple types of attacks, but there are still spikes in the number of attacks shown for some attack categories. Shenzhen Rf-Link Technology has a relatively lower amount of overall attacks.

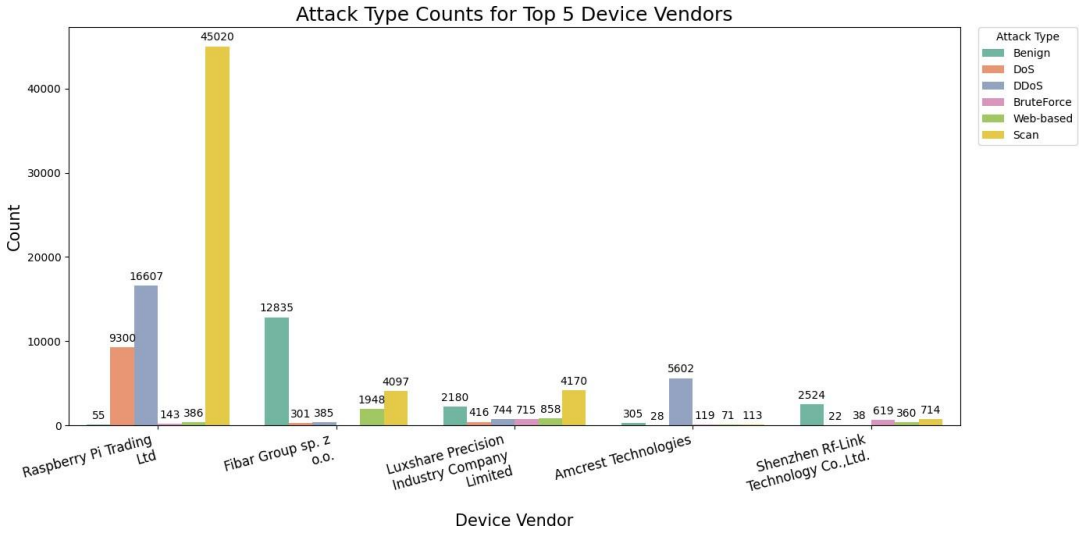

Figure 5: Attack Type Distribution Across Top 5 IoT Device Vendors

The treemap in Figure 6 demonstrates the relationship between device vendors, the type of attack, and the network protocol used in the attack vector. Each rectangle's size represents the number of records and is grouped first by vendor, then by attack type, and lastly by protocol. Raspberry Pi Trading Ltd leads the chart, with the majority of the traffic consisting of Scan, DDoS, and DoS attacks using TCP. Fibar Group shows a large portion of benign traffic but still has Scan and web-based attacks in TCP/UDP. Vendors such as Luxshare Precision Industry Limited and Amcrest Technologies show a mix of attack types and protocols, while lesser-known vendors provide a more scattered distribution.

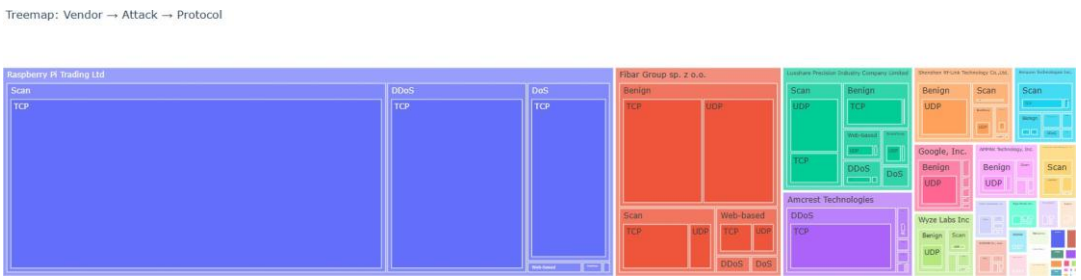

Figure 6: Treemap of Device Vendors by Attack Type and Protocol

Box plots in Figure 7 indicate how active mean (top) and idle mean (bottom) differ across attacks. In the active mean plot, DoS attacks show the largest median value,

occupying a range of values between 5M–15M, but exhibiting some extreme outliers that exceed 110M. DDoS attacks were also larger, where active mean values were found mainly in the medians of 10M–12M. The benign value is mostly less than 5M, although some outliers exhibit values that exceeds over 15M. Traffic categorized as BruteForce, Web-based, and Scan is typically under 5M of active means.

In the idle mean plot, surprisingly, the benign traffic has the widest spread, where this range is also 10M–50M, but outliers exhibit values around 115M. DoS and DDoS had sizeable idle means where the majority were between 5M–20M, with some exceeding 100M. The idle variability is higher for web-based and brute force, compared to the active mean; this may indicate the lack of burst activity, with more periods of pause from burst activity.

Figure 7: Active and Idle Mean Distributions by Attack Type

The histogram in Figure 8 indicates the distribution of packet length variance over a variety of attack types. The majority of values for all types are heavily concentrated on the low end (under 1M variance) across all types of attacks, but there are differences in count. Scan traffic has the most occurrences, with over 100k in the low variance range, followed by Benign and DDoS traffic, with 40k-50k occurrences in the low variance range. DoS has the next fewest occurrences, around 20k. Bruteforce and web-based attacks appear in even fewer occurrences. Values with a variance above 2M are uncommon in all categories, but there are outliers as high as almost 13M. Taken together, these data suggest that most network traffic, malicious or not, typically has stable values for packet sizes, but when there are few examples of traffic that has values of high variability, it usually indicates abnormal communication or bursty communication.

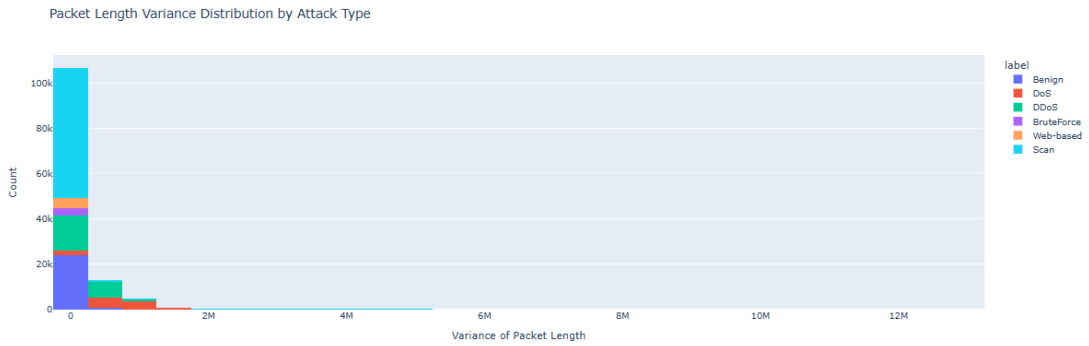

Figure 8: Packet Length Variance Distribution by Attack Type

3.4 Feature Engineering

During the feature engineering stage, we began with the dataset's dimension. Initially, the dataset is found to be very large, with 125,519 rows and 204 columns, so we wanted to understand its size and complexity. Then we extracted several time-based features from the timestamp column, which we first converted to a proper datetime object to isolate its components and create time, hour, minute, day of week, weekend, and working hour indicator components for the timestamp column. Time-based features help to capture time-based behavioral patterns in network traffic. The attacks typically happen more during certain hours or days. In turn, these patterns can greatly increase predictive power. Then we removed columns that have only a single unique value, like some of the protocol flags, since they do not provide any variability and therefore cannot be learned by the model. Next, we checked for the categorical features to consider their uniqueness and determined that there were fields with high cardinality (such as IP addresses) and low variance fields that we could discard. We encoded all the categorical variables, including the target label, using label encoding so that they could be used for advanced deep learning models, and the original datetime column was removed after it was exploded because it was now redundant.

To find the most significant variables, we trained a Random Forest model and plotted the cumulative feature importance curve as shown in Figure 9. This curve illustrates the cumulative contribution of features as we increase the number of variables included in the model. Each point on the curve represents the contribution of the most important features at that point. The red dashed line represents the 95% contribution threshold of the feature importance, which can be seen as the point in which including more features contributes little to the overall ability to make predictions. The chart shows that the 95% contribution threshold is about 88 features, which indicates that the top features alone can account for the 95% predictive ability of the model. This allowed us to keep only the most predictive features, lower dimensionality, and improve model performance. The next challenge was to address class imbalance in the dataset.

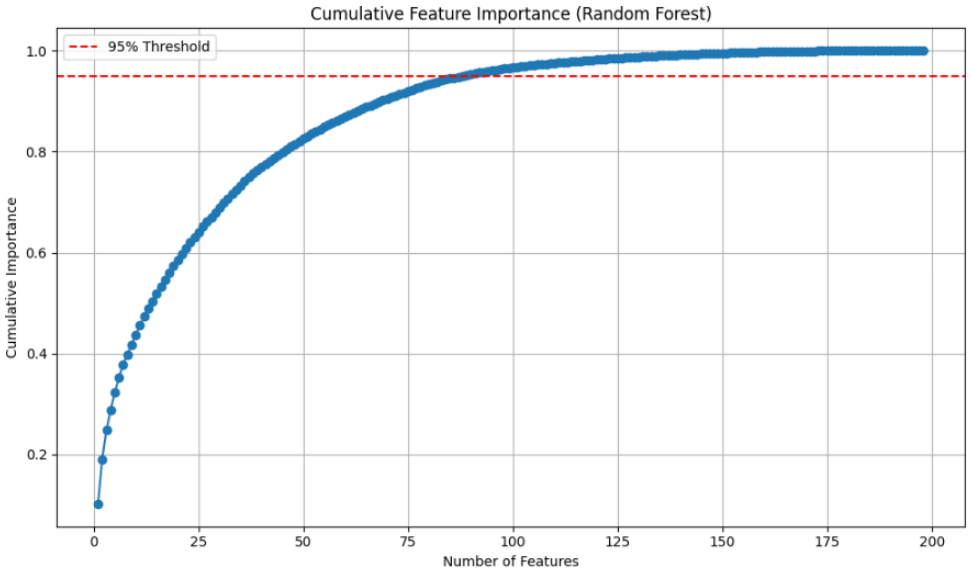

Figure 9: Cumulative Feature Importance Curve using Random Forest Model

From a class distribution analysis of the dataset, it was clear that the data was highly imbalanced, with 57,974 samples in the majority class (Scan) and only 3,505 samples in the minority class (DoS), which could impede our model's predictive accuracy. To fix the class imbalance, we applied the Synthetic Minority Oversampling Technique (SMOTE), which synthesizes new samples from the minority classes. Once we applied SMOTE, all classes were balanced to 57,974 samples, allowing for complete representation from all classes and better detection of all attack types by the model. The graphs in Figure 10 and Figure 11 summarize the class distribution before and after applying SMOTE technique.

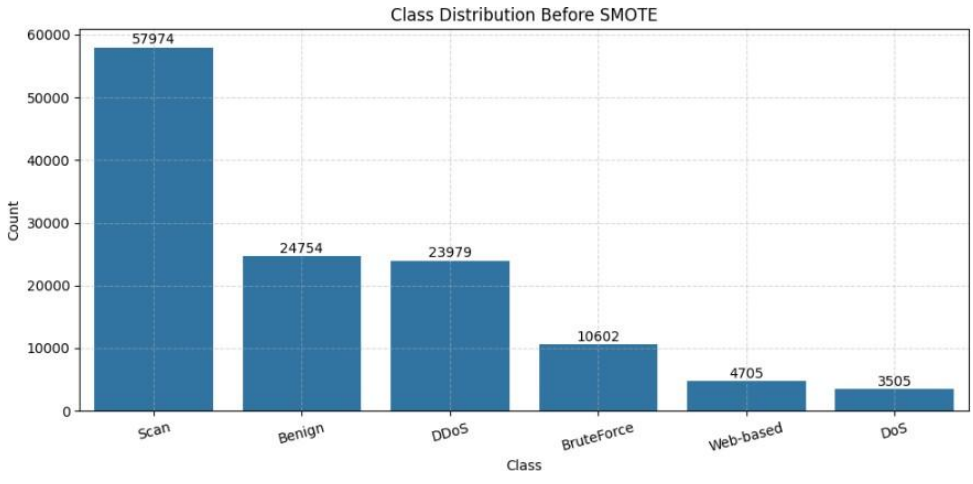

Figure 10: Class Distribution Before Applying SMOTE

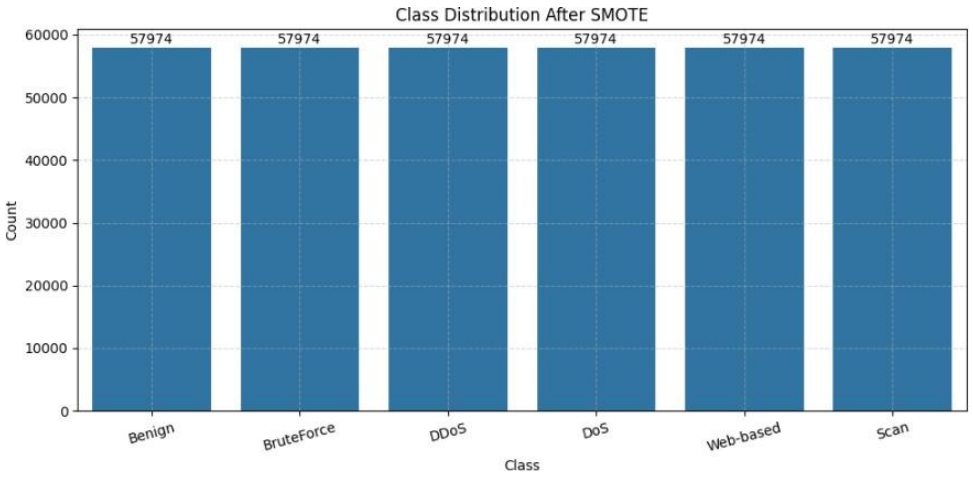

Figure 11: Class Distribution After Applying SMOTE

The dataset was then split into training and testing sets for the final model. We standardized the numeric features with StandardScaler for uniform scaling and one-hot encoded our target variable for multi-class classification. The entire feature engineering and preparation process helped reduce extraneous noise, balanced the distribution of the class components, and ensured we eliminated attributes that contained the most useful information to ensure the best model fit, which helped us to assure the attack prediction process was accurate and unbiased.

3.5 Model Training

Model training is one of the crucial steps of feeding the prepared data into a machine or deep learning algorithm in order for it to discover the patterns, relationships, and decision boundaries that help it to make accurate predictions on unseen data. It is an important part of this research as we aim to build an intelligent intrusion detection system that reliably classifies different types of IoT attacks on computer networks. In this study, we trained three different deep learning models: Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM), and our innovative hybrid Convolutional Long Short-Term Memory (Conv-LSTM) model. The Conv-LSTM model is our proposed architecture that combines the feature extraction capability of CNNs and the sequential learning ability of LSTMs for capturing spatial and temporal dependencies present in network traffic data over time. The dataset was divided with a 70% training and 30% testing split to ensure every model receives an opportunity to learn and also evaluate based on unseen samples of data. These specific models are selected because CNNs can automatically extract local features from complex data, LSTMs are ideally suited to learning long-term temporal dependencies needed when considering network traffic analysis as it happens over time, and the Conv-LSTM model utilizes both the benefits of CNNs and LSTMs, making it an excellent candidate for recognizing and detecting a variety of continued and evolving IoT attacks.

3.6 Model Evaluation

Model evaluation is the evaluation of trained model performance on test data to see how well it has learned on the training data to make accurate predictions. Evaluation is an important step because it determines if the model is only fitting the training data or it can generalize effectively on new, unseen real-world inputs. In this research since target label has multiple types of attack, we consider this as a multiclass classification problem. All three models, Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM), and Convolutional Long Short-Term Memory (Conv-LSTM), will be evaluated with the key performance metrics such as accuracy, precision, recall, and F1-score. Each model's loss value has been tracked and recorded during every epoch of training to observe the learning behavior of the model. We included a confusion matrix for all models to give further insight into the model performance for correct and incorrect predictions across all classes. All the performance evaluation results were visually represented through graphs, charts, and plots clearly and effectively for direct comparison among models. The detailed results analysis and discussion of the results will be described in the results section.

4 Design Specification

This section covers the design specifications of the deep learning models proposed in our research. Each model's architecture, underlying principle, and implementation decisions are discussed for the development of an application for IoT intrusion detection.

4.1 Convolutional Neural Network (CNN)

The Convolutional Neural Network (CNN) is deep learning architecture broadly used in situations involving structured grid data, such as images, signals, or time-series, because it relies solely on convolution operations to automatically learn valuable feature sets. In general, CNNs are good at detecting spatial patterns, and can find localized features without requiring manual feature engineering. In this research, we applied this CNN model on 1D network traffic data, and started the model with a convolutional layer utilizing ReLU activation, as this enables the model to discover local patterns in the input features. This was followed by batch normalization to stabilize and accelerate the training, dropout layers to mitigate the risk of overfitting, and a max pooling layer to reduce dimensionality while maintaining the more impacting features. We then flattened the feature maps and continued with fully-connected (dense) layers, with the final output layer being softmax for multiclass prediction. In this research, CNN is implemented, as it is very efficient at learning the spatial relationships of features, which is a requirement for recognizing attack patterns contained within an IoT network. The architecture of the Convolutional Neural Network is shown in Figure 12.

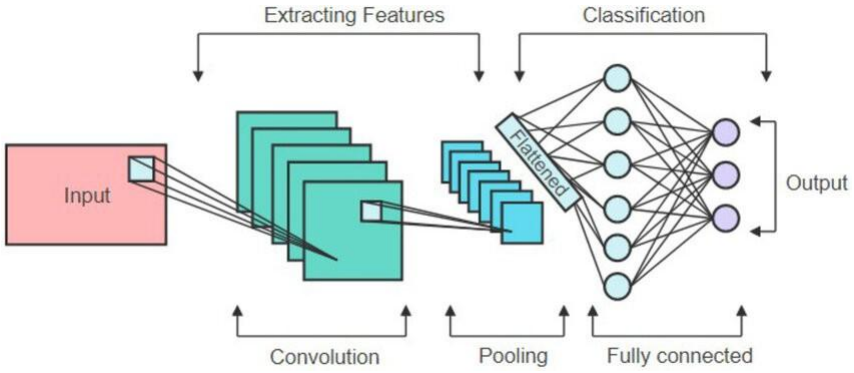

Figure 12: Architecture of Convolutional Neural Network Nagy et al. (2023)

4.2 Long Short Term Memory (LSTM)

The Long Short-Term Memory (LSTM) network is a form of recurrent neural network (RNN) that has the within-framework ability to learn from sequential or time-related data, while overcoming the issues inherent to regular RNNs, mainly the vanishing gradient issue. LSTM networks are known for their ability to remember long-term dependencies while being able to forget unimportant information via a gating mechanism. In this research, the LSTM model is selected to adopt the temporal dependencies and sequences of IoT network traffic. The architecture begins with an LSTM layer to learn long-term temporal patterns, followed by a batch normalization layer and a dropout layer for model generalization benefits. The next layers are the dense layers that perform functions on the established features, and a softmax layer is placed on the last layer to output the class probabilities. LSTM is chosen because many IoT attacks leverage a significant temporal signature, such as repetitiveness of request patterns or consistently malicious behavior, making the memory characteristics of LSTM very relevant for more accurate detection. The architecture of the Long Short-Term Memory (LSTM) is shown in Figure 13.

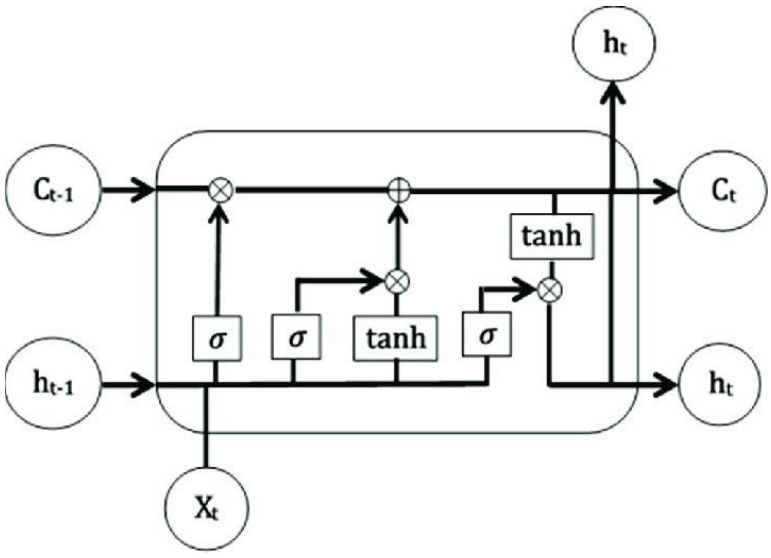

Figure 13: Architecture of Long Short Term Memory Toma et al. (2019)

4.3 Convolutional LSTM (Conv-LSTM)

The Convolutional Long Short-Term Memory (Conv-LSTM) architecture is a hybrid model that successfully integrates the process of spatial feature extraction with the ability to learn from temporal sequences. The Conv-LSTM model can be utilized for spatio-temporal data, like video data, or sequential sensor reads from IoT data where timeliness and spatial patterns are both necessary. We proposed Conv-LSTM as a new method for IoT network intrusion detection. The architecture begins with time-distributed convolutions layers to extract spatial features independently from each timestep then passes that data through a batch normalization and dropout layer to enable stable and robust learning. The features derived at each timestep are then connected to an LSTM layer to ensure we extract temporal dependencies across the entire sequence. Then the LSTM output (the combined timesteps) is passed into dense layers to analyze a series of spatial-temporal features, and from the final dense layer to a softmax layer to develop multiclass predictions. The hybrid framework for the proposed model was optimal since CNN's capacity only parallelly analyzes spatial patterns and LSTM's capacity only serially analyzes temporal dependencies and the Conv-LSTM architecture can exploit both of these processes, while also having to resolve the complexity and high dimensionality of IoT data when detecting an intrusion in a timely manner. The architecture of the Convolutional LSTM is shown in Figure 14.

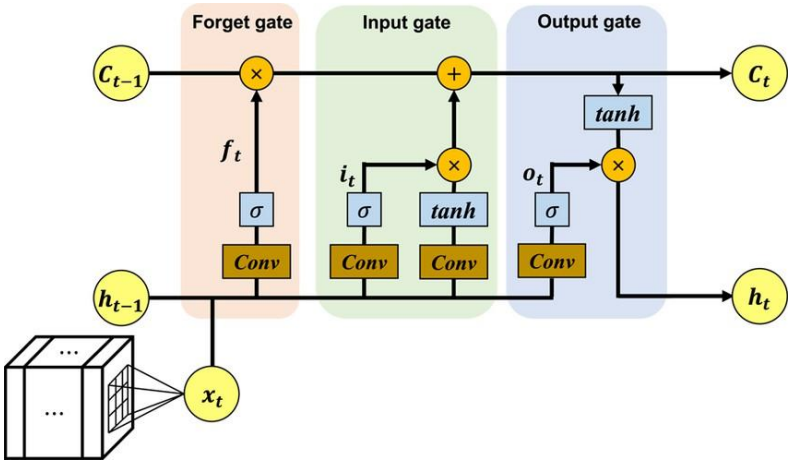

Figure 14: Architecture of Conv-LSTM Jeong et al. (2024)

5 Implementation

The implementation of this research utilized Python, a high-level programming language, due to its usability and readability. We imported standard libraries and modules such as numpy and pandas for numerical computations and managing dataset operations, textwrap for formatting output files, Counter from the collections module for counting and summarizing class distributions, and joblib for efficiently reading and writing trained models. We also used libraries for visualization such as matplotlib and seaborn for creating static plots, and plotly for interactive visualizations, primarily for exploring and understanding data distributions. The scikit-learn library was also heavily relied upon for data prep and evaluation since it provided functions and classes that we needed for

processes such as imputation of missing values (SimpleImputer), splitting the dataset into train-test, encoding target labels and scaling input features (LabelEncoder and Standard Scaler), generating ensemble measures of importance (RandomForestClassifier), and evaluation metrics (classification_report and confusion_matrix). Scikit-learn also provides many models and utilities to tackle imbalanced datasets. We use the Synthetic Minority Oversampling Technique (SMOTE), from the imbalanced-learn library, to help create synthetic samples in minority classes to ensure our model was trained fairly. We intended to develop and train our models primarily using TensorFlow/Keras, where we implemented Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM), and Convolutional LSTM architectures using the available Keras base layers. We selected Adam as our optimizer because it allowed us to optimize efficiently. We also implemented EarlyStopping to prevent overfitting by observing the validation loss value. To convert categorical features to numbers, we used label encoding, and we used L2 regularization to improve generalization to observations. Importantly, we managed to integrate signal processing functions as necessary (fft and fftfreq from the scipy.fft module) to allow us to explore frequency-domain analysis.

We extended our work beyond building and evaluating our models and developed a real-time IoT intrusion detection web application using HTML/CSS and Flask to deploy our best-performing model from our research. Our application continually collects IoT network traffic and utilizes an ML/DL model to classify incoming packets in real time, displaying the attack type, packet ID, confidence level, and action taken, such as triggering the IDS alert. As shown in Figure 15, the application includes elements such as a Live IoT Device Packet Timeline, showing currently detected activity, and other metrics of interest, such as the attack distribution pie chart, bar chart for attack category count, and live attack timeline line graph depicting total activity and activity by category over time. Using these visualizations, users can use the application for fast interpretation of the security state of their network, and can take action as required.

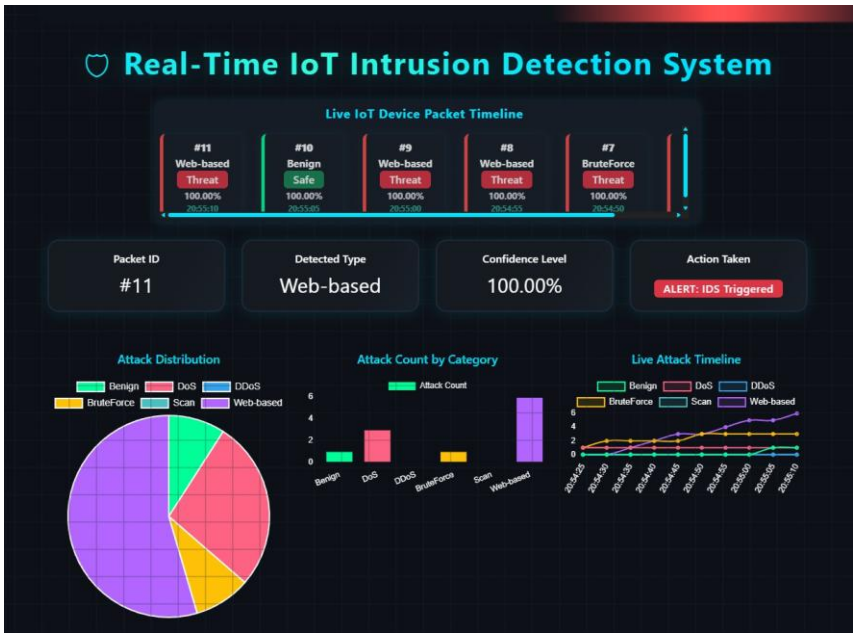

Figure 15: Real-time IoT Intrusion Detection System

6 Evaluation

This section presents in-depth evaluation of the proposed models based on many performance metrics. The purpose of the evaluation stage is to quantitatively measure the effectiveness of each architecture for accurately detecting and classifying IoT network intrusions. Once each metric is evaluated, we will see both the advantages and limitations of each model, and can decide which model is most appropriate for real world implementation.

6.1 Experiment-1 / Evaluation based on Accuracy Score

We evaluated the performance of our IoT intrusion detection models based on accuracy, which is a key metric of interest. Accuracy is a measure of the proportion of correct predictions against all predictions made by the model used. It is a valued metric, as higher values signify better detection. For this research, we compared the three models Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM), and the proposed hybrid Convolutional LSTM (Conv-LSTM) using the same test dataset. The CNN model has achieved an accuracy of 97.40%; on the other hand, LSTM achieved a greater accuracy of 99.73%, and the Conv-LSTM accuracy delivered the best performance with an accuracy score of 99.99%. This establishes that while both CNN and LSTM presented very strong models, the hybrid Conv-LSTM model utilizes the spatial feature extraction ability of CNN while concurrently using the temporal sequence learning capabilities of LSTM to deliver near-perfect results.

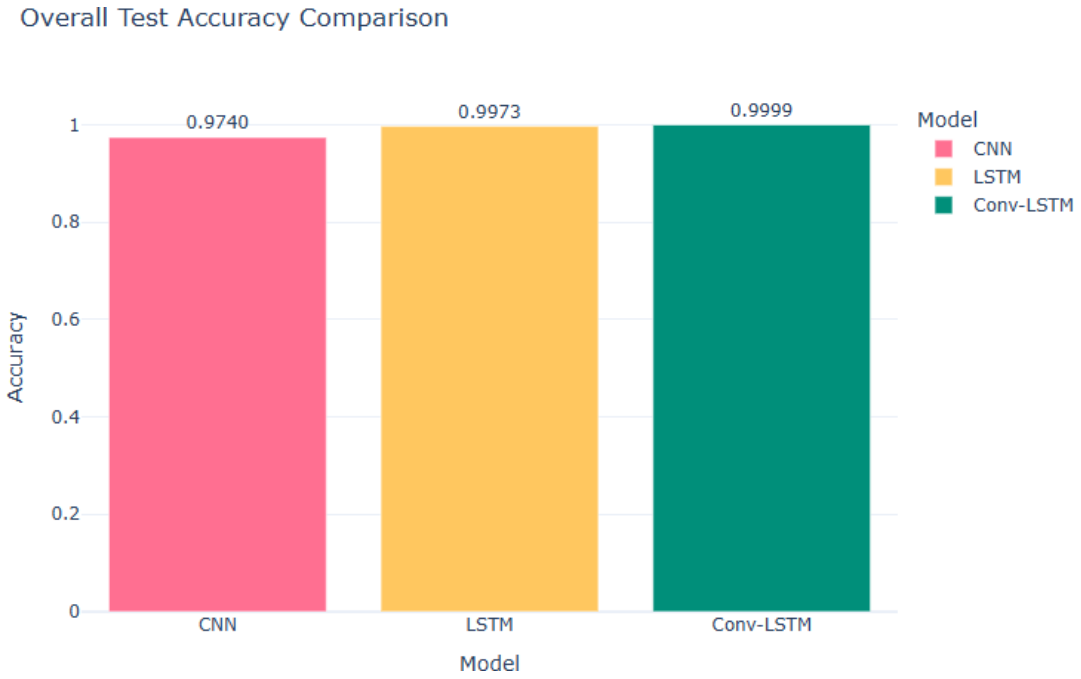

Figure 16: Accuracy Comparison of CNN, LSTM and Conv-LSTM Models

6.2 Experiment-2 / Evaluation based on Precision Score

Precision is a measure of the proportion of positively predicted instances that are correct out of the total number of instances predicted as positive. In terms of detection, it tells us how well the model is identifying attack instances, not only attacking instances, but also how few false positives there are. A higher precision value is better because it will have fewer incorrect alerts for attacks, improving the model's reliability to generate and notify alerts. As per our analysis, the CNN model had a macro-averaged precision of 0.9770, while both the Long Short-Term Memory (LSTM) and hybrid Convolutional Long Short-Term Memory (Conv-LSTM) models had a precision score of 0.99. Our results showed LSTM models had better performance to correctly capturing the attack instances, causing fewer alerts, while not classifying or misclassifying benign traffic.

Figure 17: Precision Comparison of CNN, LSTM, and Conv-LSTM Models

6.3 Experiment-3 / Evaluation based on Recall Score

Recall is a metric that measures the number of true positives correctly identified out of the actual positives present in the dataset. Recall can measure an intrusion detection model's capability to identify all existing attacks while minimizing false negatives. The higher the recall score, the better the system is at detecting malicious activity accurately. In this case, the macro-averaged recall score for the Convolutional Neural Network (CNN) model was 0.9740, the Long Short-Term Memory (LSTM) model was 0.9973, and the hybrid Convolutional Long Short-Term Memory (Conv-LSTM) model achieved the highest macro-averaged recall score of 0.9999. These results indicate that all models provide reliable intrusion detection abilities; however, the Conv-LSTM model was better at capturing a large proportion of observed attack instances accurately.

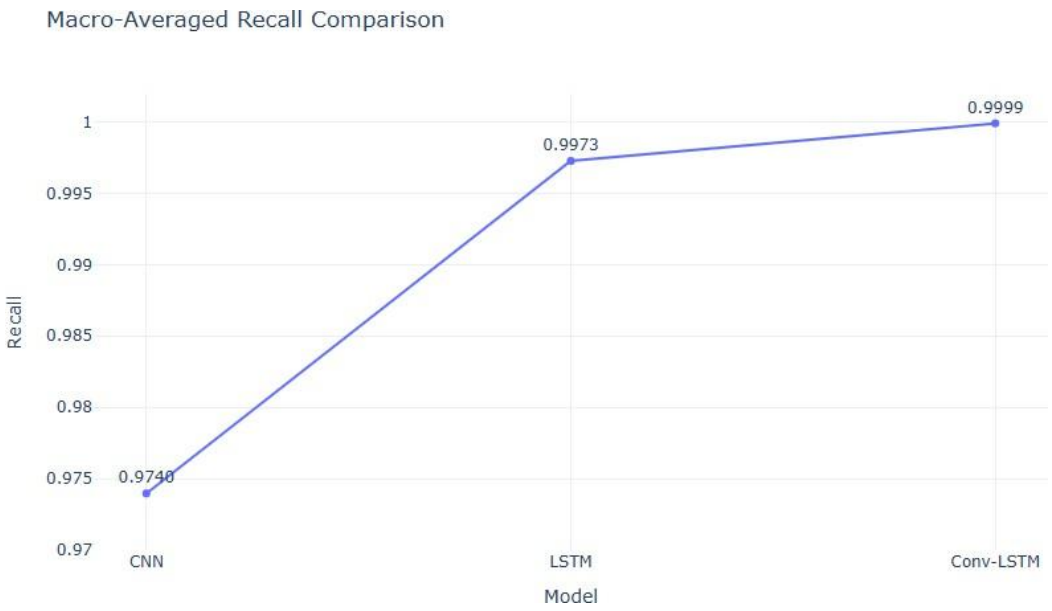

Figure 18: Recall Comparison of CNN, LSTM and Conv-LSTM Models

6.4 Experiment-4 / Evaluation based on F1-Score

F1-score provides the harmonic mean of precision and recall with a balanced measurement of both false negatives and false positives. F1-score is significant when assessing models for intrusion detection systems, where low recall (missed detections) and low precision (false alarms) have serious consequences. The higher the F1-score is, the better the model performs in correctly identifying attacks and false alarms. In our experiments, the convolutional neural network (CNN) model produced a macro-averaged F1-score of 0.9743, followed by the long short-term memory (LSTM) with 0.9973, and the Conv-LSTM achieved the highest score rate of 0.9999. This outcome highlights that the Conv-LSTM model has the best confirmation of balance for detection accuracy and non-detection error among the models.

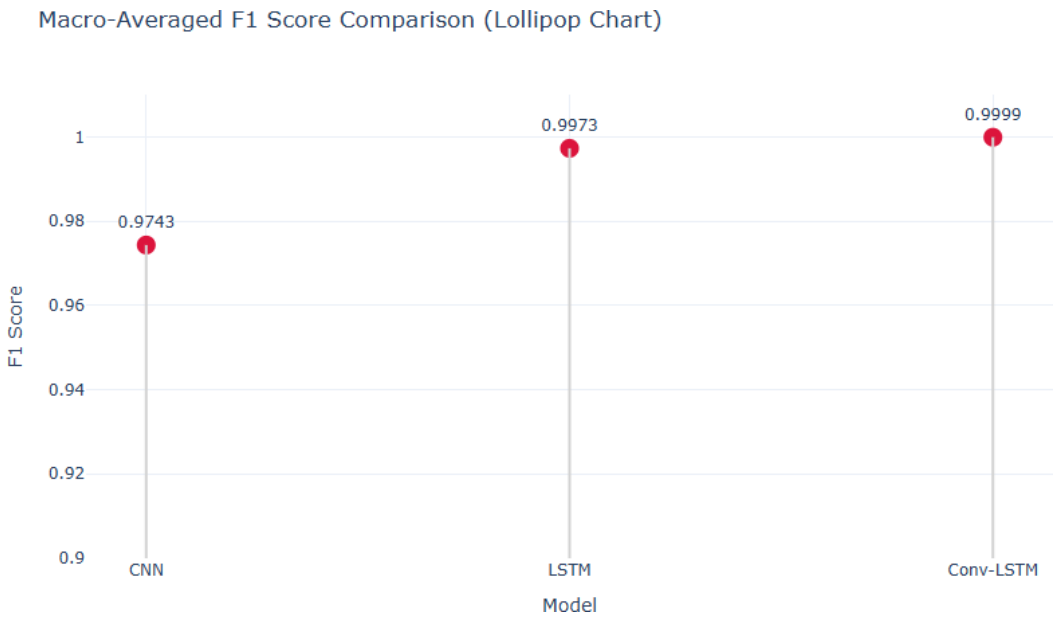

Figure 19: F1-Score Comparison of CNN, LSTM and Conv-LSTM Models

6.5 Discussion

The experimental results show a clear separation of the performance among Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM), and Convolutional Long Short-Term Memory (Conv-LSTM) models. These models were evaluated based on accuracy, precision, recall, and F1-score. In all evaluation metrics, Conv-LSTM significantly outperformed CNN and LSTM. Conv-LSTM achieved scores of 0.9999 for every metric. LSTM achieved scores close to Conv-LSTM, while CNN trailed LSTM in recall and F1-score. The difference in model performance is likely related to model architecture. CNN can effectively capture spatial patterns, but neither generates a model that can account for temporal sequence. Thus, it is not the best choice for analyzing sequential IoT traffic. LSTM can account for sequential time-series data and therefore, generate a long-delayed dependency model, but it lacks the types of local feature extractive properties that CNN can employ. The Conv-LSTM architecture integrates both model's advantages; extracting high-level spatial features through the CNN convolutional layers, while simultaneously modeling long-term dependencies through the LSTM layers. Therefore, Conv- LSTM has better detection performance.

A more in-depth assessment of the training characteristics of the Conv-LSTM indicates that both the training accuracy and validation accuracy increased rapidly during the training process before stabilizing at values nearly equal to 1.0 within a few epochs. Likewise, the loss values dropped quickly and remained very low throughout the training process indicating the learning and generalization by Conv-LSTM was strongly effective as shown in Figure 20.

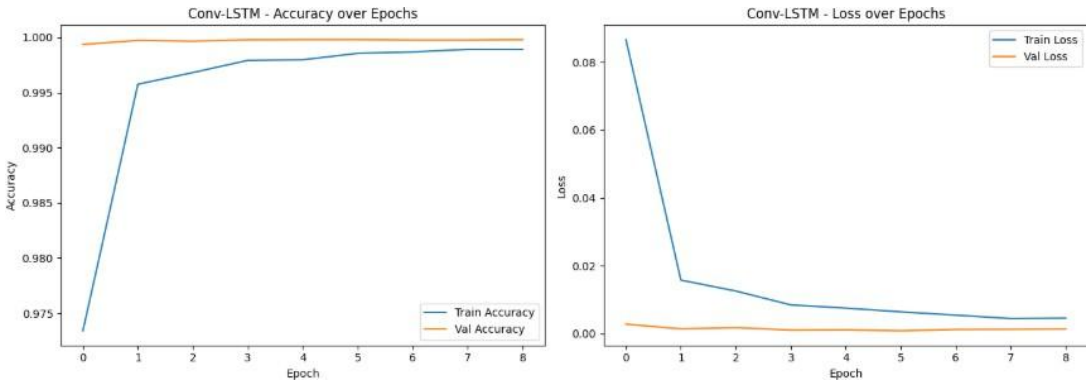

Figure 20: Accuracy and Loss Score of Conv-LSTM Model over every epoch

The confusion matrix for Conv-LSTM as shown in Figure 21 further confirmed this performance for classification as shown with a high degree of representation for all attack type categories: Benign, Denial of Service (DoS), Distributed Denial of Service (DDoS), BruteForce, Scan, and Web-based attack with almost perfect classification, and a very small number of misclassifications (e.g. only 6 BruteForce samples were misclassified). Since the classification performance achieved such high precision and recall for the attack type categories, it provides further confidence to conclude that implementing Conv-LSTM in a real-time IoT intrusion detection application remains a robust choice. Overall, choosing Conv-LSTM justifies high accuracy and delivers a reliable degree of generalization performance to unseen data through balanced and effective exploitation of the spatial and temporal features in the computing platform.

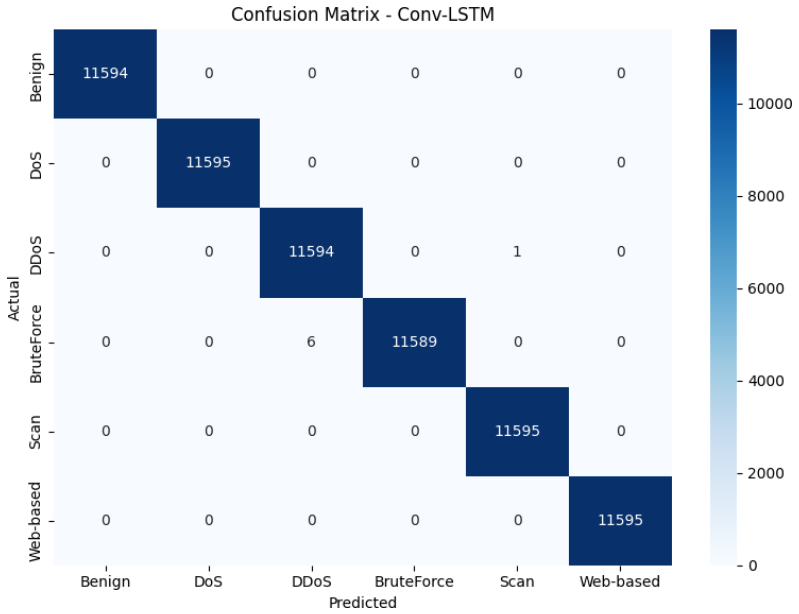

Figure 21: Confusion Matrix of Conv-LSTM Model

7 Conclusion and Future Work

In this research, we have successfully developed and evaluated an advanced real-time IoT intrusion detection system utilizing three deep learning architectures, CNN, LSTM, and Conv-LSTM on the IoT-IDAD 2024 dataset. The three architectures were compared across accuracy, precision, recall, and F1-score with the results indicative that the Conv-LSTM architecture produced the best performance and the Conv-LSTM model was near perfect in all metrics. The hybrid architecture of the model allows for the scene characterization of CNN and spatio-temporal sequence modeling of LSTM to allow for precise detection of all kinds of IoT attack patterns, including DoS, DDoS, BruteForce, Scan, and web-based attacks. The real-time web application of the proposed model further demonstrated the practical applicability of the system with immediate detection and alerts. Overall, the Conv-LSTM based IDS proposed in this research presents a powerful, scalable, and highly accurate solution to protect IoT environments from ever-evolving cyber threats.

The proposed Conv-LSTM-based IoT intrusion detection system performed exceptionally well under controlled circumstances, however, future research may help build upon this work by extending the system's capabilities for real-world, large-scale IoT deployments. For example, the architecture could be extended with online learning mechanisms to enable the model to adapt to the continuously changing attack patterns without the need for complete model retraining. Another possibility would be the implementation of federated learning, thereby enabling the system to allow cooperative training of the model across multiple distributed IoT networks while preserving the privacy of the local data resources of individual IoT networks. In an effort to better defend against newly created and zero-day threats, robust detection methods could be explored further that combine deep learning methods with rule-based anomaly detection strategies.

References

Ahmad, Z., Khan, A. S., Nisar, K., Haider, I., Hassan, R., Haque, M. R., Tarmizi, S. and Rodrigues, J. J. P. C. (2021). Anomaly detection using deep neural network for iot architecture, *Applied Sciences* **11**(15): 7050.

Apostol, E. S., Truică, C. O., Pop, F. and Esposito, C. (2021). Change point enhanced anomaly detection for iot time series data, *Water* **13**(12): 1633.

Chatterjee, A. and Ahmed, B. S. (2022). Iot anomaly detection methods and applications: A survey, *Internet of Things* **19**: 100568.

Cook, A. A., Misirli, G. and Fan, Z. (2020). Anomaly detection for iot time-series data: A survey, *IEEE Internet of Things Journal* **7**(7): 6481–6494.

Diro, A., Chilamkurti, N., Nguyen, V. D. and Heyne, W. (2021). A comprehensive study of anomaly detection schemes in iot networks using machine learning algorithms, *Sensors* **21**(24): 8320.

Gaddam, A., Wilkin, T., Angelova, M. and Gaddam, J. (2020). Detecting sensor faults, anomalies and outliers in the internet of things: A survey on the challenges and solutions, *Electronics* **9**(3): 511.

Giannoni, F., Mancini, M. and Marinelli, F. (2018). Anomaly detection models for iot time series data, https://arxiv.org/pdf/1812.00890.

Haji, S. H. and Ameen, S. Y. (2021). Attack and anomaly detection in iot networks using machine learning techniques: A review, *Asian Journal of Research in Computer Science* pp. 30–46.

Hasan, M., Islam, M. M., Zarif, M. I. I. and Hashem, M. M. A. (2019). Attack and anomaly detection in iot sensors in iot sites using machine learning approaches, *Internet of Things* **7**: 100059.

Jeong, S.-H., Lee, W., Kil, H., Jang, S., Kim, J. and Kwak, Y.-S. (2024). Deep learning-based regional ionospheric total electron content prediction—long short-term memory (lstm) and convolutional lstm approach, *Space Weather* **22**.

Mukherjee, I., Sahu, N. K. and Sahana, S. K. (2023). Simulation and modeling for anomaly detection in iot network using machine learning, *International Journal of Wireless Information Networks* **30**(2): 173–189.

Muntean, M. V. (2024). Real-time detection of iot anomalies and intrusion data in smart cities using multi-agent system, *Sensors* **24**(24): 7886.

Nagy, W., Alsalamah, H., Hassan, M. and Moahamed, E. (2023). Auto-har: An adaptive human activity recognition framework using an automated cnn architecture design, *Heliyon* **9**: e13636.

Nizam, H., Zafar, S., Lv, Z., Wang, F. and Hu, X. (2022). Real-time deep anomaly detection framework for multivariate time-series data in industrial iot, *IEEE Sensors Journal* **22**(23): 22836–22849.

Priyadarshini, I., Alkhayyat, A., Gehlot, A. and Kumar, R. (2022). Time series analysis and anomaly detection for trustworthy smart homes, *Computers and Electrical Engineering* **102**: 108193.

Rabbani, M., Gui, J., Nejati, F., Zhou, Z., Kaniyamattam, A., Mirani, M., Piya, G., Opushnyev, I., Lu, R. and Ghorbani, A. A. (2024). Device identification and anomaly detection in iot environments, *IEEE Internet of Things Journal* .

Sahu, N. K. and Mukherjee, I. (2020). Machine learning based anomaly detection for iot network: (anomaly detection in iot network), *Proceedings of the 4th International Conference on Trends in Electronics and Informatics (ICOEI)*, pp. 787–794.

Sgueglia, A., Di Sorbo, A., Visaggio, C. A. and Canfora, G. (2022). A systematic literature review of iot time series anomaly detection solutions, *Future Generation Computer Systems* **134**: 170–186.

Sharma, B., Sharma, L. and Lal, C. (2019). Anomaly detection techniques using deep learning in iot: A survey, *Proceedings of 2019 International Conference on Computational Intelligence and Knowledge Economy (ICCIKE)*, pp. 146–149.

Toma, R., Nahid, A. and Hasan, M. N. (2019). Electricity theft detection to reduce non-technical loss using support vector machine in smart grid.

Ullah, I. and Mahmoud, Q. H. (2021). Design and development of a deep learning-based model for anomaly detection in iot networks, *IEEE Access* **9**: 103906–103926.

Yang, M. and Zhang, J. (2023). Data anomaly detection in the internet of things: A review of current trends and research challenges, *International Journal of Advanced Computer Science and Applications (IJACSA)* **14**(9).
URL: *http://www.ijacsa.thesai.org*