Detection of Security Vulnerabilities in IoT Devices using Advanced Deep Learning methods within Cloud Computing Framework

MSc Research Project
MSc in Cloud Computing

Himavanth Raavi
Student ID: X23101083

School of Computing
National College of Ireland

Supervisor: Aqeel Kazmi

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Himavanth Raavi
Student ID:	X23101083
Programme:	MSc in Cloud Computing
Year:	2023-2024
Module:	MSc Research Project
Supervisor:	Aqeel Kazmi
Submission Due Date:	03/01/2025
Project Title:	Detection of Security Vulnerabilities in IoT Devices using Advanced Deep Learning methods within Cloud Computing Framework
Word Count:	2996
Page Count:	20

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:	Himavanth Raavi
Date:	29th January 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):	

Detection of Security Vulnerabilities in IoT Devices using Advanced Deep Learning methods within Cloud Computing Framework

Himavanth Raavi
X23101083

Abstract

The rapid proliferation of IoT devices, along with cloud computing capabilities, has made disruptive changes to modern industry by enabling seamless connectivity, scalabilities, and data processing. While this incorporation gave rise to many advancements, it, however, raised some of the most critical vulnerabilities from a security perspective, as IoT systems are becoming vulnerable to synthetic malware attacks. Traditional security mechanisms, usually static and rule-based, always have limitations in identifying and mitigating these complex threats, especially in real-time scenarios of dynamic IoT-cloud environments. This research proposed a deep learning solution to tackle problems with regard to malware detection in IoT systems hosted in the cloud environment. Upon implementing and evaluating three deep learning models based on accuracy, precision, recall, and F1-score among CNN, RNN, and auto encoders, we have identified CNN as the top model, consistently outperforming as compared to the other models across all metrics. This trained CNN model was deployed in a cloud-based web application running on an AWS EC2 instance for real-time monitoring and classification of network traffic. It comes with a user-friendly interface to allow the classification of traffic into benign or malicious in order to address the threats on time to the system administrators. The proposed framework successfully provides detection and handling of bulk IoT traffic and addresses important security challenges by leveraging the strength and availability of the cloud.

1 Introduction

Internet of Things (IoT) has become a network trend in recent years that embedded several devices to use communication services introduced within the Internet protocols. Considerably, these devices are smart objects that operate anonymously in an environment with no direct interaction among humans, thus serving as a distinct component with unique functionalities. Over the years, there has been a continuous surge in device connectivity through IoT networks worldwide. In 2020, approximately 21.7 billion devices were actively connected globally, which indicates that IoT devices have surpassed non-IoT connections and are expected to rise more by 2025 (30 billion approx.) Pourrahmani et al. (2023). With this shift, contemporary security challenges in the network system have been identified, continuously affecting device connectivity and activities. The growth of the IoT industry has reported a statistic of $49.04 billion in an estimated period of

2022-2026 with an accelerated CAGR rate estimated to be $27.48 billion Pourrahmani et al. (2023).

The above configuration implies an outstanding growth of the IoT market and the underlying security vulnerability that can be expected. This security challenge has been identified with the evolution of cyber-attacks, which enhances the adaptability and intricacy of the hacking process to pursue the target objectives. This issue is more concerned with the integration of IoT in the cloud environment, thus requiring a secure storage and processing system for data. While privacy is the biggest concern with secure data handling in the IoT-integrated cloud environment several detection methods have been introduced including traditional methods and machine learning algorithms. Some potential solutions such as "wireless sensor networks" (WSN), web personalization, and "Radio Frequency Identification" (RFID) have been used to identify unknown objects and enhance data protection. However, IoT is considered to be an ever-changing aspect that has encompassed various technologies while exhibiting changing features. Thus, security challenges are prominent due to the low detection accuracy of existing models.

The current study has gained motivation from this issue to explore enhanced security systems or technologies such as deep learning methods that have gained tremendous attention in recent times. Deep learning is a promising approach with its extravagant applications, especially in the detection and prevention of cyber attacks within the cloud environment for IoT devices. Hence, this study has contributed focus on investigating security vulnerabilities in IoT devices using deep learning models within the cloud computing framework. In this research project, we built a Cloud-based IoT malware detection framework. Three deep learning models were trained on IoT traffic dataset concerning the other two models-RNN and Autoencoder stating their performances against each other in detail with standard metrics. Besides, we hosted a Flask application on the AWS EC2 instance to implement the real-time monitoring and detection of malicious IoT traffic while giving room to scale in size and efficiency at the cloud level.

1.1 Research Objectives

- To Train and evaluate Convolutional Neural Network (CNN), Recurrent Neural Network (RNN), and Autoencoder models to identify the most effective approach for detecting malicious activities in IoT networks.
- To Build a scalable, real-time cloud based web application that detects IoT Malware in real-time.
- To Utilize cloud resources, specifically an AWS EC2 instance, to process large-scale IoT traffic efficiently.

1.2 Research Question

- How can deep learning models be effectively utilized to detect IoT malware in a cloud-based environment, addressing challenges such as real-time detection, scalability, and accurate classification of network traffic, while ensuring a user-friendly deployment through a web application?

2 Related Work

The contextualisation of information has been presented through empirical evidence gathered in this chapter. This approach has outlined extensive work of other researchers and scholars by reviewing evidence on security vulnerabilities typically identified in IoT networks and privacy challenges in managing highly unstructured data within the cloud environment. This empirical evidence would subsequently provide comprehensive insights into existing models and a research scope for the need to further introduce efficient detection methods.

2.1 Cyber Threat in IoT Networks

With the revolution of the network environment with IoT, a significant shift in the network system has been identified. According to the evidence presented by Aldhaheri et al. (2024) by 2025, an expected rise in IoT-connected devices to 30 billion has been determined. This forecast even though is a promising solution toward further technology-based improvement, is a boon to the industrial landscape, data breaches and privacy risks in IoT devices are major concerns. One of the potential reasons for this issue is the limitation in computational power as well as storage capacity in IoT devices, which enable hackers to invade the network system with IoT emergence more aggressively Aldhaheri et al. (2024). Industrial IoT as explained by Dhirani et al., (2021) is a concept that creates a connected, transparent, and automated network environment that has improved industrial applications through a smart approach to business processes and work efficiency. However, with this novel emergence comes challenging prospects including network security risk characterized by exploitation of network security in IoT devices and data thefts. Dhirani et al. (2021) explained that a convergent approach to understanding the malicious activities in IoT-based network systems, especially within the cloud environment is an ongoing research due to emerging features of IoT and cyber security standards that are needed.

According to the information provided by Gupta et al. (2019) state-of-the-art countermeasures within the IoT-enabled smart-grids have become an evolutionary trend to address security challenges. Cyber security risks within IoT-based smart grids or other linked devices develop malicious activities causing data espionage and physical damage to connected devices, thus aiming for exploitation and financial gains. The issue is highly enhanced by user-centric behavior in IoT networks such as a focus on smart homes that require multi-featured conditions and connected devices. With a vision for these integrated systems in the near future more exclusively, the IoT system has become more heterogeneous compared to current systems, where advanced decision-making is required. In the study presented by Akatyev and James (2019), the information addressed potential concerns with user-centric network systems and described their characteristics including device and service patterns and data flows. The study performed a threat analysis to illustrate a scenario concerning the heterogenous IoT system while outlining "cyber-physical" risks. Overall, the information presented a need to explore different detection mechanisms to support user-oriented IoT networks within the cloud environment and further demonstrate a potential to prevent device exploitation.

2.2 Detection of Security Vulnerabilities Using Traditional Methods

A paradigm shift identified in conventional power distribution to an improved smart grid has leveraged the "Information and Communication Technology" (ICT). Further evidence provided by Shokry et al. (2022) explained that the application of "advanced metering infrastructure" (AMI) has been considered a critical component in smart grids (SG). The relevance of the component is based on collecting, processing, and transferring data across the Internet. While operational benefits have marked a suitable recognition for SC, privacy challenges have emerged as a significant issue. The above study therefore presents countermeasures with the application of artificial intelligence (AI) including "decision trees" (DT), "rule-based", and "data mining" techniques. Upon leveraging these techniques, the AMI system has marked a suitable smart grid approach to mitigating security challenges by uncovering anomalies and presenting statistically-significant structures. In another study presented by Shahriar and Zulkernine (2022), the information presented certain facts regarding security vulnerabilities and the necessity for program security when leveraging the novel IoT network infrastructure. The study provides a comparative analysis of mitigation works, mapping the efficiency of different traditional techniques typically used for the detection of network anomalies.

Security violations in IoT devices have received extensive attention from digital investigations and academic researchers to understand the drawbacks of existing detection techniques. According to the information provided by Pistoia et al. (2017) a focus on static analysis to identify security vulnerabilities has become progress that uncovers definite solutions to detect security challenges in various software systems. With the progression of gathering the study findings, it has been identified that static analysis has become an emerging trend for detection and access control approaches for many years, which can be categorized as stack-based and role-based controlling accessibility. Based on the research result, it has been identified that the technique has achieved significant attention due to its performance accuracy although more research is necessary to understand the viability in an advanced IoT environment. Internet usage has become a major technological trend in the daily transactions of organizations worldwide. The information provided by Aslan et al. (2023) explained that emerging technologies for example, IoT, cloud computing and wireless communications are increasing security concerns within cyberspace. While understanding the concern, researchers and investigators have focused on detection methods that can detect advanced cyber attacks such as DDoS, phishing, and other malicious intrusions.

The innovation of feasible methods of detection has been introduced by this study, which shows that both technical & non-technical solutions have earned recognition from experts. Some common measures include blockchain, big data-based mining techniques, and spam identification. Chander and Gopalakrishnan (2019), further presented information on similar progress where WSN has gained widespread recognition in the technology realm. With higher responsiveness to sensor technologies with IoT framework that connects or links a range of devices, data privacy, and breaches have predominantly retained enhanced security upgradation to address concurrent solutions. The application of blockchain technology has made progress in the detection of cyber attacks in IoT systems, given that it monitors the integrity within real-world dataset sharing and supports fiscal transaction recording in the case of encoded connections. Understandably, disruption or modification in the occasional transaction is easily recognized, thus providing a feasible detection process with greater IoT security.

2.3 Detection of Security Vulnerabilities Using Machine Learning Methods

While traditional techniques have modeled relevant progress to detect security challenges in previous IoT infrastructures, modern systems with advanced IoT features demand improvement in cyber security. Advanced attacks such as DDoS fail to be detected by traditional security systems, therefore identifying the need for modern means Aslan et al. (2023). In the IT industry, software, and network security challenges are common concerns among developers and business organizations in the contemporary environment. Malicious hackers are continuously modifying attack features due to which existing detection methods have been facing issues in identifying the underlying threats Chernis and Verma (2018). While understanding this, Hanif et al. (2021) introduced the application of machine learning (ML) algorithms in the vulnerability detection of both software and network system. The detection significance of ML methods is based on their classification process from included datasets. With the modern IT infrastructure, supervised ML models have become a trend that has predicted advanced security attacks for network systems and provide an accuracy level of 95% with varying datasets. Waqas et al. (2022) explained that the modern IoT system has introduced both opportunities and challenges to mankind where data sharing across interconnected devices is a reflecting part. Based on the information demonstrated by the study, it is understandable that the advanced interconnectedness among IoT devices is one of the reasons for security vulnerabilities.

Undeniably, machine learning classifiers have presented significant opportunities in the detection process and studies have extensively provided insights into this aspect. With the advancement in cyber attacks, security vulnerabilities in the network system for IoT devices have threatened the technological approach across multiple industries. In the study presented by Naveen and Sharma (2022), the information has illustrated the function of IoT-enabled devices, which has gained recognition in the recent decade. While the use of connected devices has reached 20-21 billion, a further rise is expected in upcoming years. This has created significant concern among researchers and experts regarding security challenges, which would also experience a hike. Therefore, the above study has presented insights into the prediction of advanced attacks using pre-process steps in data processing to be applied in classifiers. A suitable approach to feature selection, classification of data, and detection of issues from evaluating these features is a common mechanism of ML models. Alwahedi et al. (2024) explained that the rapid advancement in ML techniques has achieved improved performance in the prediction of security challenges compared to traditional models. This information even though has been explained repetitively by many studies, the implication provides distinct insights with in-depth analysis of each study. Findings established from the above study show that the contribution of the information is valuable to understanding the inherent security issues and the continuous improvement needed in potential security measures.

2.4 Detection of Security Vulnerabilities Using Deep Learning Methods

The continuum established through an informational approach to understanding IoT security vulnerabilities has presented evidence of different detection mechanisms. Previously different traditional state-of-the-art methods and advanced machine learning classifiers have been presented through systematic reviews of their performance in the detection of network attacks. However, the increasing complexities in the attack features have challenged previous models with not-so-reliable performance. Therefore, the integration of deep models in practice has gained significant attention. According to the information provided by Aversano et al. (2021), the introduction of deep neural architectures has gained momentum in recent years due to the undeterred consistency in the detection of growing cyber-attacks. The implication of the above study has further explained the challenges faced by traditional ML models in the prediction and regression of features from selected datasets. Aversano et al. (2021) further explained that the growing network architecture based on DL models has involved security aspects containing features, highlighting underlying security threats. Supporting this information, another study presented by Khan et al. (2022) has exclaimed the extensive growth of deep learning models due to their relevance in the detection of vulnerabilities in computer networks.

In the above study, it has been explained that the detection of intrusions in IoT-based data generation (big data) is extremely challenging due to which existing models have under-validated their performance significance. Although the previous evidence on supervised ML models shows that decision trees and random forests have presented higher performance accuracy, the increasing complexity and size of exponential data from IoT networks require advanced mechanisms. As such, deep models integrated with IDS systems are critically evaluated based on their performance using different parameters such as accuracy rate, precision, recall rate, and F1-score. Indicating the experimental outcome, Khan et al. (2022) demonstrated a greater significance of DL models compared to existing models. IoT is an innovative technological advance that has interconnected devices worldwide. A wide understanding of this phenomenon has been presented in the literature, and it has uniquely identified the global significance of this interconnection. However, the information is constantly disclosed by studies concerning realms of discussion on network security challenges with the advancement of the IoT system.

In the study presented by Ullah et al. (2019), the demonstration provided knowledge of the need for software privacy as well as identifying malware and network threats. In this regard, the study has introduced a "TensorFlow" deep learning model, preferably, a neural architecture that shows potential in identifying pirated software through "source coded plagiarism". This security intervention is enhanced by using the "Google Code Jam" (GCJ) dataset, thus providing higher performance, which is indeed better than the existing outcomes. Further evidence presented in another study by Al-Garadi et al. (2020) has introduced similar aspects of technological progress through IoT advancement. Amid this focus, the study has introduced the purpose of a combined Ml-DL intervention in providing security solutions to the network facing security vulnerabilities. Based on the outcomes gathered from the study, it has been identified that a hybrid technique sets a new paradigm in the detection process. At the same time, it presents future implications to identify its potential in addressing complex cyber-security challenges.

3 Methodology

Our methodology focuses on detecting IoT malware in a cloud computing environment using the UNSW-NB15 dataset Moustafa and Slay (2015). We will be performing various steps including data preprocessing, exploratory data analysis (EDA), and feature engineering to prepare the dataset for training and also perform model evaluation. Each step of our methodology is required to achieve our research objectives. Figure 1 shows the methodology flow diagram for IoT Malware detection.

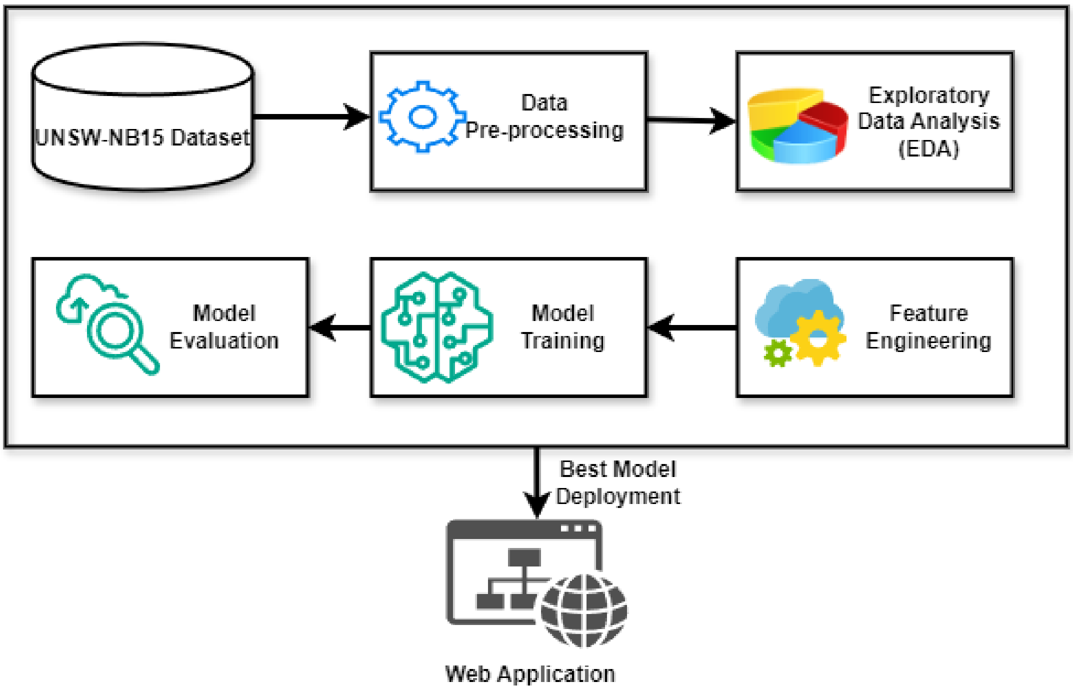


Figure 1: Methodology Flow Diagram for IoT Malware Detection

3.1 Dataset Description

The UNSW-NB15 dataset consists of four CSV files with a total size of around 559 MB. Each of the files has around 700000 rows and 49 columns Moustafa and Slay (2015). The IXIA PerfectStorm tool created this dataset in the Cyber Range Lab at UNSW Canberra., simulating a mix of real-world normal network activities and synthetic attack behaviors. The dataset contains records of some such attacks, including Fuzzers, Analysis, Backdoors, DoS, Exploits, Generic, Reconnaissance, Shellcode, and Worms, as well as details of the network flow for each attack. The dataset features 49 very important fields that represent the behavior of network traffic, and each record is tagged to indicate whether it is a normal activity or an attack. This dataset will be used in this research to identify and mitigate security vulnerabilities in IoT environments.

3.2 Data Pre-processing

Data preprocessing ensures that the dataset is clean and ready for analysis. This research did essential data preprocessing steps to clean the dataset. We checked for missing values in our data, and we found that ct_flw_http_mthd, is_ftp_login, and attack_cat contained missing values. Specifically, ct_flw_http_mthd had 1,348,145 missing values, is_ftp_login had 1,429,879, and attack_cat had 2,218,764 missing values. The imputing involved using the median for the numerical columns and the mode for the categorical column. Next, duplicate values were checked for, and 480,630 duplicate rows in the dataset were found. We then removed duplicates to avoid skewing our analysis with redundant data.

3.3 Exploratory Data Analysis (EDA)

Exploratory Data Analysis (EDA) is a vital step during the analysis and understanding of a dataset. It helps in finding hidden patterns, and anomalies, assumptions testing, and in summarizing the key characteristics of the data. In this research, we will apply EDA using various charts and graphs to analyze the dataset, understand its structure, and highlight meaningful patterns.

The chart in Figure 2 consists of a bar chart and a pie chart. They represent the distribution of the "Label" column which states whether a particular traffic is benign (0) or attack (1). The bar chart on the left indicates that benign traffic (Label 0) constitutes the majority in this dataset, with over 2 million records, while attack traffic (Label 1) is very small, nearly 100,000 records. The pie chart on the right reaffirms this conclusion, with 95.2% labeled "benign" and 4.8% labeled "attack". The bar chart, along with the pie chart, clearly shows the class imbalance

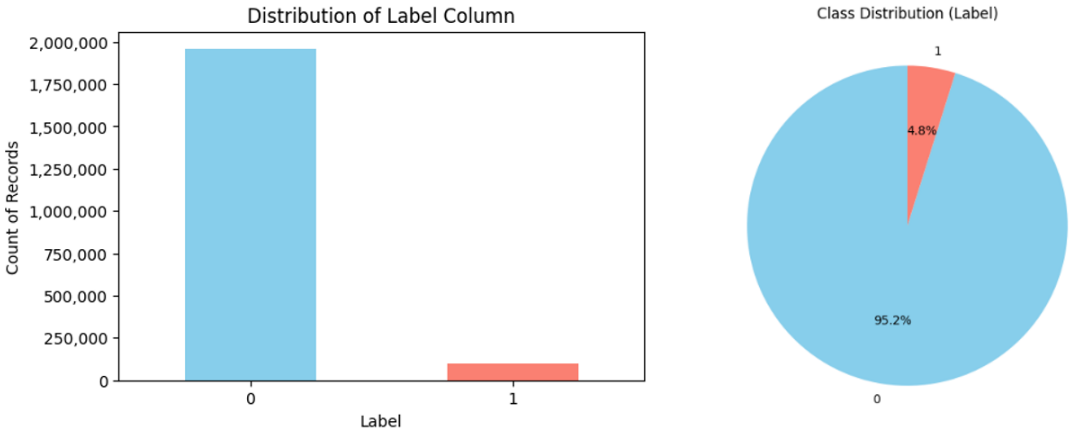


Figure 2: Distribution of Label Column (Benign vs Attack Traffic)

The following graphs in Figure 3 show the log-transformed distribution of bytes from source to destination and packet counts from source to destination. The left graph indicates benign traffic, denoted as Label 0, with the larger byte values, and the attack traffic, denoted as Label 1, more spread out across smaller values. While conversely, in the right graph, benign traffic again has a higher number of packet counts, while attack traffic concentrates more on lower packet counts. The benign traffic, Label 0, in numbers gives around 2 million records, while the attack traffic, Label 1, gives around 100,000 records.

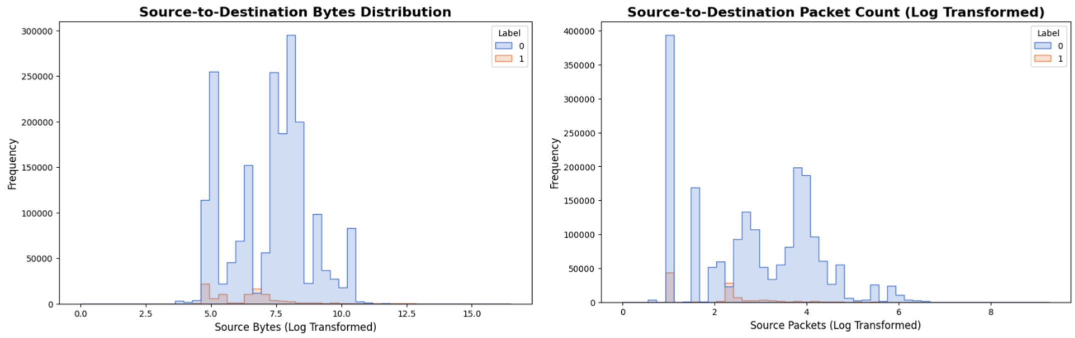


Figure 3: Distribution of Source-to-Destination Bytes and Packets (Log Transformed)

The strip plot shown in Figure4 represents the total traffic across several attack categories and shows that Exploit and DoS attacks dominate in total traffic with Exploit traffic peaking at around 14 million records. In contrast, Fuzzers, Worms, Backdoors, and Shellcodegenerated comparatively low traffic; however, there are a few outliers in the data.

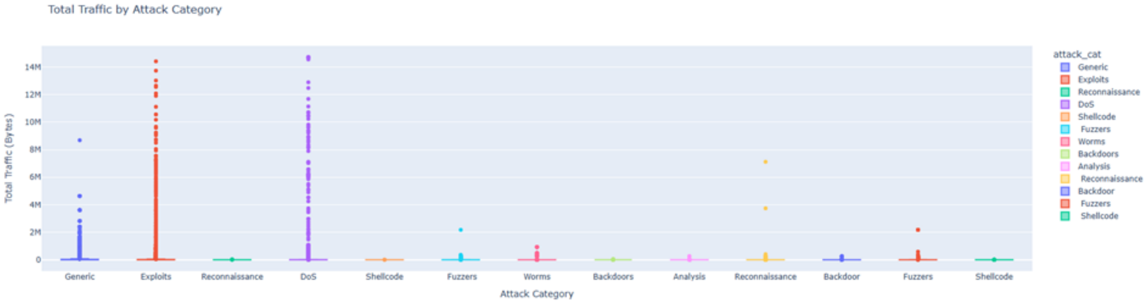


Figure 4: Total Traffic by Attack Category

The box plot in Figure 5 represents the average packet size across various attack categories. It indicates that the Exploit and DoS attacks feature a higher mean packet size, the former showing a large variation and numerous outliers. On the other hand, Generic, Recon, and Fuzzers feature lower mean packet sizes. The intention is to illustrate the variation of packet sizes across different attack categories, which helps to identify kinds of attacks that may put a heavy burden on network resources or that have uniquely distinguishable packet characteristics.

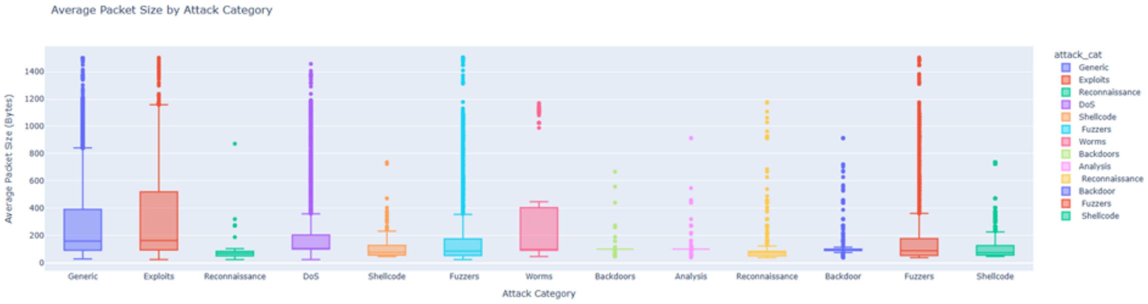


Figure 5: Average Packet Size by Attack Category

The radar chart in Figure 6 compares network features like duration, source bytes, destination bytes, source packets, and destination packets across several attack types. The chart shows that Fuzzers display high values for the source bytes and destination bytes, demonstrating dramatic spikes for several features. Worms exhibit greater duration and number of packets, while Exploit and DoS attacks could be discerned by their superior values for source packets and destination bytes. Reconnaissance and Backdoors usually come out with lower values, showing variations in traffic characteristics related to those attack categories.

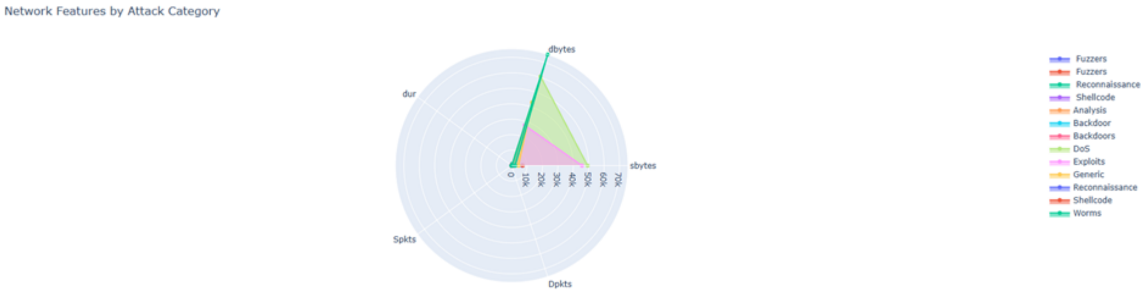


Figure 6: Network Features by Attack Category

A sunburst graph in Figure 7 displays a representation of benign (Label 0) and attack (Label 1) traffic. The data show that 95% of the traffic concerns benign service (Label 0), while only a smaller proportion accounts for attack traffic (Label 1). Within benign service, DNS and HTTP were 18% and 9%, respectively. The chart shows that services that are commonly used in normal and attack services can be identified, revealing that most traffic pertains to different benign services while attack traffic, in general, is limited to fewer services.

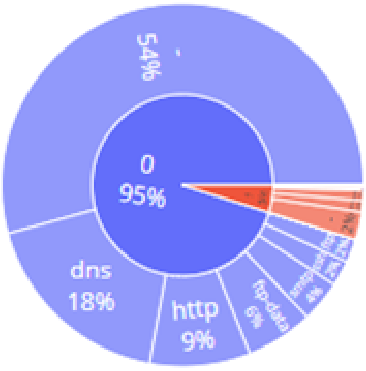


Figure 7: Service Distribution by Benign vs Attack Traffic

The Sankey diagram in Figure 8 represents the benign and attack traffic flowing through various services. The bulk of the benign traffic is said to flow through services such as FTP, SMTP, and HTTP, while the attack traffic is concentrated on the FTP-data, SNMP, and SSL services. This gives a good visual for identifying which services are actively associated with benign and the attack-type traffic, demonstrating potential attack vectors and further areas for attention in security monitoring.

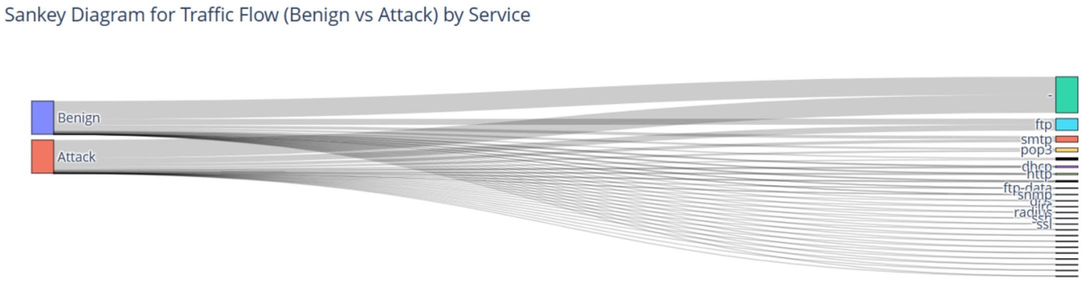


Figure 8: Sankey Diagram for Traffic Flow (Benign vs Attack) by Service

The bar chart in Figure9 represents the top 10 source and destination IPs based on the number of records in the dataset. The blue indicated the source IPs while the red indicated the destination IPs. It appears that IPs such as 59.166.0.4, 59.166.0.1, and 149.171.126.3 would quite often appear as both the source and destinations with nearly equal counts in either role. This graphic allows identifying the most active source and destination IPs, indicating patterns of network traffic that could potentially present interests in an analysis of security monitoring or traffic analysis.

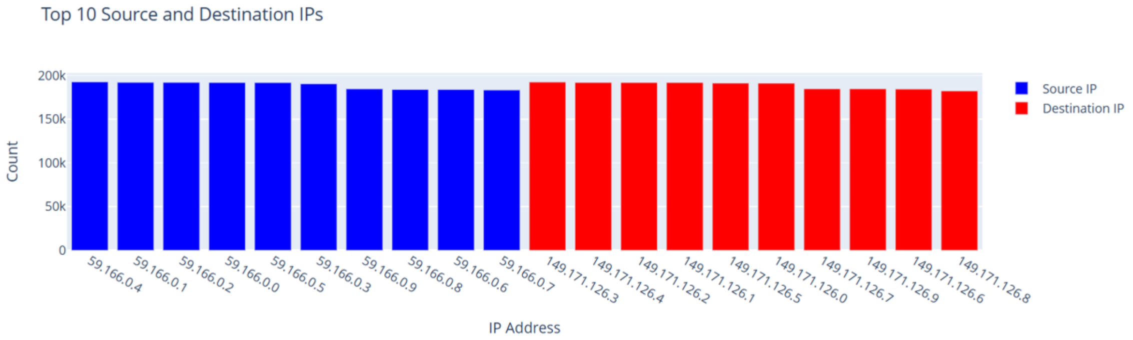


Figure 9: Top 10 Source and Destination IPs

3.4 Feature Engineering

Feature engineering is the process of changing the raw data into meaningful features to enhance model performance. To identify highly correlated features ($threshold > 0.8$) that might induce redundancies, we began by calculating and visualizing the correlation matrix. Based on this description, we removed highly correlated features, such as dbytes, Dpkts, and ackdat, to make the data set efficient. We implemented Label Encoding to convert the categorical text features into numerical forms appropriate to model training. To resolve the class imbalance in the target variable, namely Label, we performed the SMOTE Under Sampling method to obtain a balanced dataset by modifying the number of samples of each class as shown in Figure 10. By this process, better quality datasets with less redundancy in adequate distribution for training the model are gained.

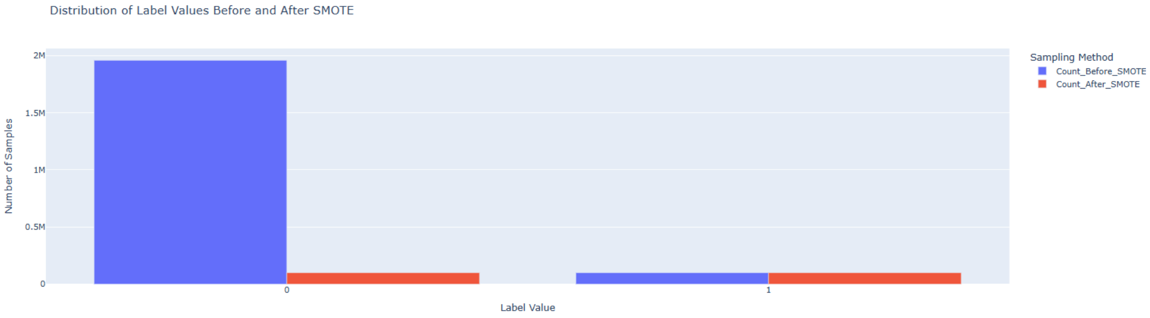


Figure 10: SMOTE to Handle the Imbalanced Data

3.5 Model Training

This research aims to compare three models, namely CNN, RNN, and Autoencoder, to find the most effective for IoT malware detection in the cloud environment. CNNs or Convolutional Neural Networks are selected due to their known efficiency in capturing spatial patterns in data, which helps identify attack signatures that have complex design in terms of their flows in network traffic. RNNs or Recurrent Neural Networks are chosen since they possess great strength in processing sequence-based data, capable of capturing temporal dependencies in network traffic for indicating the possible existence of malware actions. The third Autoencoder is selected explicitly based on its capability of anomaly detection, as an autoencoder would provide great functionality in this regard since it can model normal traffic very well, while detecting outliers or anomalies is necessary to capture all classes of new or unseen malware. Through the comparison of these three models, we want to discover which approach will provide the best results in terms of accuracy and efficiency for IoT malware detection.

3.6 Model Evaluation

During this research, performance evaluation of the CNN, RNN, and Autoencoder models is conducted based on various metrics. Model evaluation is thus an important aspect since it indicates how the models are capable of generalization on unseen data, signifying that the resultant model is capable of detecting malware with fewer false positives and false negatives. Models will be evaluated based on various performance metrics including accuracy, precision, recall, and F1-score, giving a full-fledged view of each model's capability to efficiently classify malware and benign traffic correctly. The accuracy will give an overall impression of correctness, but precision and recall will provide adequate insight about the model's capability to identify malware while preventing benign traffic from being misclassified, and vice versa. The F1-score gets averaged together and thereby balances precision and recall, providing a single measure of model performance. Also, calculation of the confusion matrix will be done to visualize the assessment of models, one can detect at what point misclassification has happened. Model evaluation process will help us to identify the most optimal model for IoT Malware detection.

4 Design Specification

The figure 11 describes how the IoT malware detection system works in a cloud environment. IoT devices include smart sensors, cameras, and connected systems that generate network traffic data, such as packet metadata, communication protocols, and usage patterns. The Cloud Gateway facilitates secure data transmission from IoT devices to the cloud for processing, analysis, and real-time malware detection.Once data from IoT devices is gathered, the cleaning process takes place. This typically involves filling in missing values, removing duplicates, and other techniques to achieve data consistency, thereby ensuring quality and readiness for further processes. Afterward, the processed data is run through the deep learning model, which learns the normal behaviour of network traffic and can identify the signs of malicious network activity through the learnt patterns for each type of attack. The results are then forwarded to the web application, which continuously monitors incoming traffic for any signs of malware. Once a suspected IoT malware is detected, an alerted user will trigger notifications to the administrator. Consequently, this will notify the administrator of such an event to undertake the correct course of action to deal with the incident's threat. Thus, this process allows for real-time detection and swift containment of IoT malware attacks in the cloud setup. The overall system aims to provide for continuous monitoring, rapid response times, and management of IoT-based threats with the use of deep learning within cloud infrastructure. An alerting capability together with automatic detection enables the system to remain functional and responsive in a dynamic IoT environment.

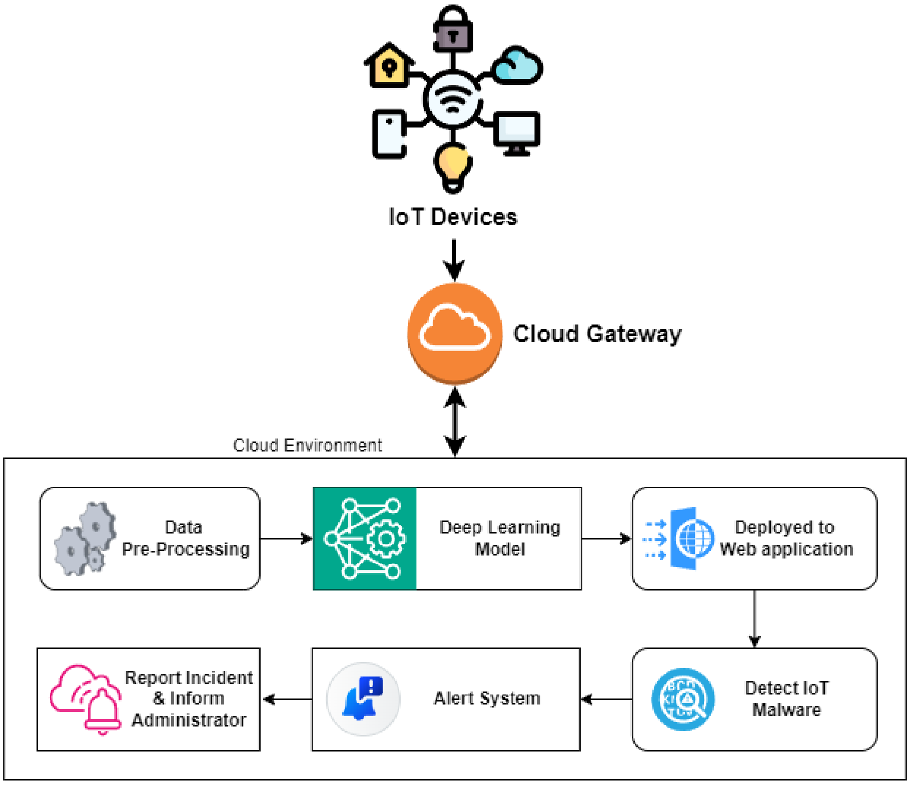
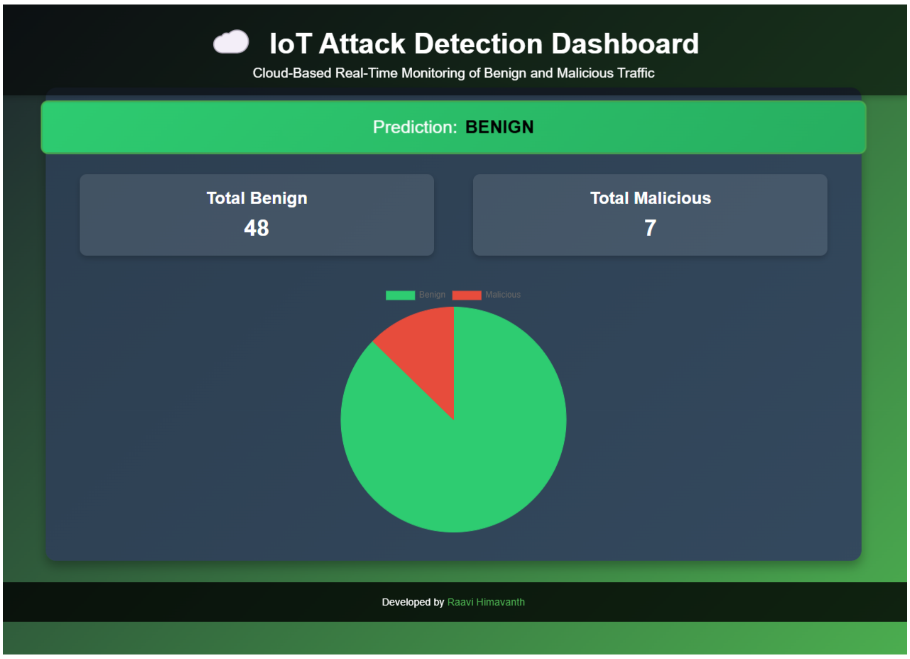


Figure 11: System Architecture for IoT Malware Detection in Cloud Environment

5 Implementation

For our research, we applied Python along with some essential libraries, such as TensorFlow, Scikit-learn, Pandas, Matplotlib, Seaborn, and Plotly, for processes ranging from preprocessing and exploratory data analysis (EDA) to model development, evaluation, and deployment. We preprocess the dataset by handling missing values, encoding categorical variables, and scaling numerical features using StandardScaler. The SMOTE UnderSampler approach corrected class imbalance through oversampling. We constructed and trained our framework on three deep learning models with TensorFlow: CNN, RNN, and autoencoders. The performance of the models was evaluated based on the measures of accuracy, precision, recall, and F-score, and the CNN was found to be the best-performing model to detect IoT malware in a cloud environment. In order to gain insights and gain a more profound understanding of the dataset, we performed profound exploratory data analysis (EDA), using Matplotlib, Seaborn, and Plotly. Various graphs and charts were created, bar plots, scatter plots, pie charts, box plots, and others. So far, through these visualizations, patterns, relationships, and anomalies within the data were identified. Model training, preprocessing, and feature engineering were conducted on Google Colab, using its computation capabilities to run the whole thing. After the model was successfully build, the web-setup was deployed on an AWS EC2 instance for real-time monitoring and asset scalability. The web application depicted in the figure 12 offers many features to enable the easy detection of malicious IoT attacks. It processes incoming network packets in real-time, classifying packets as "Benign" or as "Malicious IoT Attack." The interface dynamically updates the results including the total counts of benign and malicious packets alongside a pie chart visualizing traffic distribution. The real-time monitoring dashboard is thus crucial for proactive threat detection enabling administrators to take swift action in mitigation of IoT-related security risks. By merging deep learning with real-time alerting, we can offer our application as an easy solution to such kinds of attack log database threats in cloud security by detecting malicious IoT directly.



IoT Attack Detection Dashboard

Cloud-Based Real-Time Monitoring of Benign and Malicious Traffic

Prediction: **BENIGN**

Total Benign 48

Total Malicious 7

Developed by Raavi Himavanth

Figure 12: Web Application Detecting real-time IoT Malwares

6 Evaluation

The evaluation phase plays a very important role in getting a reliable and an efficient model deployed. The evaluation results of the three trained models, namely CNN, RNN, and Autoencoder, will be discussed in this section based on different metrics, namely accuracy, precision, recall, and F1-score, to select the most appropriate of the three trained models.

6.1 Experiment-1 / Evaluation based on Accuracy

Accuracy, a very important statistic in model assessment, is defined as the ratio of the number of correct predictions by the model over total predictions, giving an overall performance score. After performing the comparative analysis as shown with the help of bar graph in Figure13, CNN stands out with the highest prediction accuracy of 99.25%, making it very reliable for IoT malware detection. RNN, a little behind, conformed to the second-best model with a score of 98.88%, indicative of its ability to capture sequential patterns effectively. Autoencoder gave an accuracy of 98.74%, meaning it would be good in anomaly detection but lagging in some classification tasks.

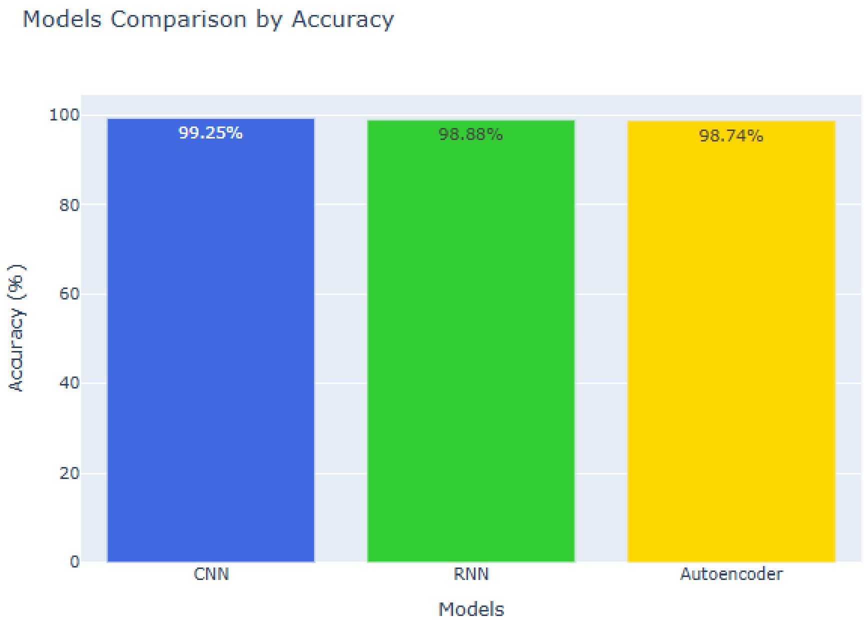


Figure 13: Accuracy Comparison of Models for IoT Malware Detection

6.2 Experiment-2 / Evaluation based on Precision

Precision is a very important measure for evaluating how well a model can identify true positives and overall predicted positives. Precision will enable an evaluation of how accurately a model detects malicious IoT traffic without raising a false alarm. The bar graph in Figure 14 represents the comparison between the precision values of CNN, RNN, and Autoencoder. The highest on the scale, RNN, achieves a precision score of 0.9917, thus being the model most reliable in minimizing false positives. Very close, Autoencoder has a value of precision at 0.9825, really a strong performer in detecting IoT malware. The CNN model earns a score slightly lower than the RNN of 0.9855, also performing well.

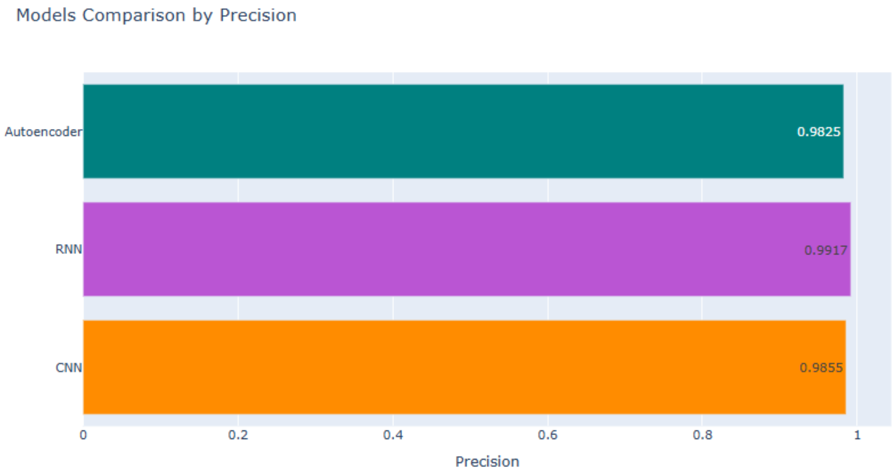


Figure 14: Precision Comparison of Models for IoT Malware Detection

6.3 Experiment-3 / Evaluation based on Recall

Recall measures a model's capacity for identifying all positive instances while minimizing false negatives. The line chart in Figure 15 compares the recall scores for CNN, RNN, and Autoencoder models. CNN has the highest recall at 0.9998, demonstrating its great capability of detecting most of the malicious IoT attacks with remarkable efficiency. Autoencoder achieves second-best recall at 0.9926, indicating that it is well-performing but slightly behind the CNN. RNN does well with a recall score of 0.9859 but is much less effective than the other two. This comparative analysis shows that CNN outperforms on recall and is hence the most reliable model for wide-ranging detection in IoT security.

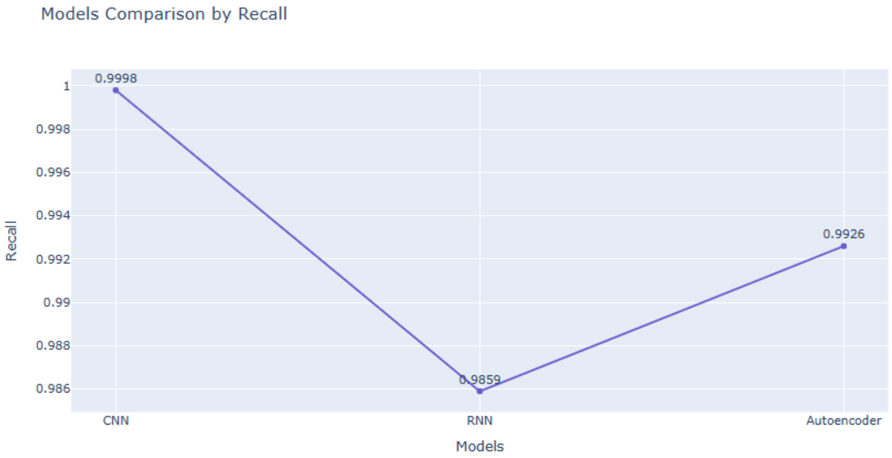


Figure 15: Recall Comparison of Models for IoT Malware Detection

6.4 Experiment-4 / Evaluation based on F1-Score

F1-score is one of the essential evaluation metrics incorporating both precision and recall to present a harmonic mean. The F1 score becomes effective in imbalanced dataset since it maintains a balance between the positives and negatives. Therefore, it's mostly relevant in the context of IoT malware detection. The radar chart in Figure16 shows the F1 scores of all three models. The model CNN displayed the highest F1 score, or 0.9925, outscoring others in the F1 scores and achieving a balanced performance between precision and recall. The following was an RNN model with an F1 score of 0.9888, which evidences effective performance but slightly lower than CNN. The Autoencoder gets an F1 score of 0.9875 and achieves solid performance while ranking third among the models. The results imply that out of these, CNN is a better-performing model, and therefore, it is more fit for deployment for real-time IoT malware detection systems.

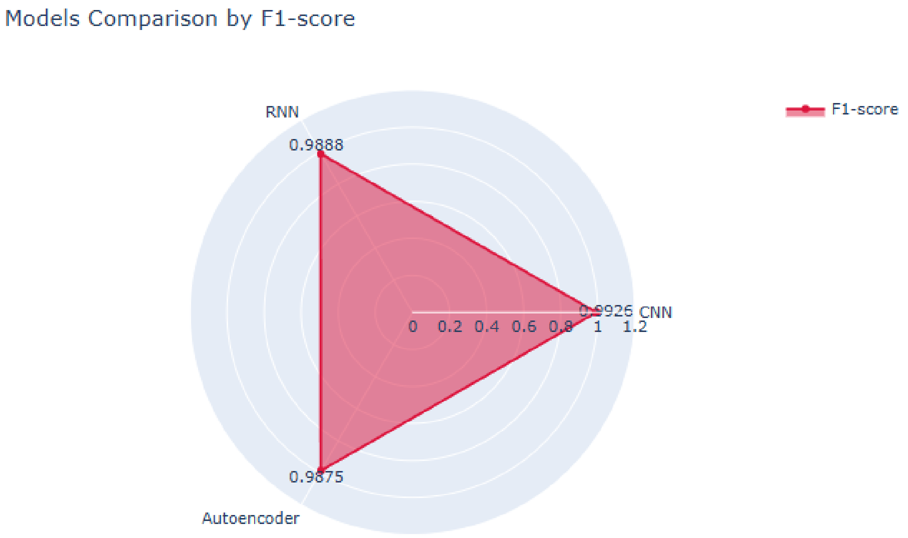


Figure 16: F1-Score Comparison of Models for IoT Malware Detection

6.5 Discussion

IoT malware detection is of utmost importance in cloud computing environments, where diverse forms of IoT traffic bring forth several significant challenges such as detection of malicious activities in real-time, dealing with imbalanced datasets, and ensuring the model is scalable. These challenges were addressed by training three models (CNN, RNN, and Autoencoder) and evaluating them on IoT traffic data. The CNN performed the best, yielding an accuracy of 99.25%, precision of 0.9855, recall of 0.9998, and an F1-score of 0.9925. This success was accredited to CNN's ability to capture the spatial feature of IoT traffic data. RNN achieved F1-score 0.9888; it can do clear sequential patterns but underperformed relative to CNN due to more computational load. Meanwhile, the Autoencoder was weak to F1-score 0.9875 because the Autoencoder focuses on unsupervised learning and anomaly detection rather than precise classification. These results show that structured CNN feature extraction makes it more suited in real-time IoT malware detection to challenges faced on cloud environments like high-volume imbalanced traffic data.

7 Conclusion and Future Work

Cloud computing is important to manage and process loads of data generated by IoT devices, in real-time, increasing scalability and usable resources. However, the increasing interconnectivity creates increased security threats such as the operational continuity of services and confidentiality of information being threatened by malware on IoT devices. This requires a powerful, lightweight, and real-time malware detection system to meet such challenges. In this research, we developed a comprehensive IoT malware detection framework utilizing three deep learning models—CNN, RNN, and Autoencoder. After extensive evaluation based on accuracy, precision, recall, and F1-score, we identified CNN as the most effective model due to its ability to extract spatial features and provide superior performance metrics. The CNN model was integrated with a real-time web application built with Flask to detect malicious IoT traffic in real-time. The web application has a simple user interface that states the predictions showing the count of benign and malicious traffic and highlighting the present detection result. The entire web application has been deployed on an AWS EC2 instance, ensuring scalability and accessibility in real-world environments. By leveraging the power of cloud computing and deep learning, our solution significantly enhances IoT security by enabling real-time monitoring and detection of malicious activities, addressing a critical challenge in modern cloud-enabled IoT systems. This research not only highlights the importance of integrating AI with cloud infrastructure but also provides a practical tool for improving the security of IoT networks. In the future, this research can be further developed by studying various advanced deep learning architectures, such as transformers and hybrid models, to improve detection accuracy and the ability to scale up in real-time environments. Deploying the system on multi-cloud environments with threat intelligence feeds would greatly improve system reliability and proactive detection capabilities. The administrators' dashboards for improved visualization and energy-efficient optimization will all contribute towards ensuring a more sustainable and user-sustained solution.

References

Akatyev, N. and James, J. I. (2019). Evidence identification in iot networks based on threat assessment, *Future Generation Computer Systems* **93**: 814–821.

Al-Garadi, M. A., Mohamed, A., Al-Ali, A. K., Du, X., Ali, I. and Guizani, M. (2020). A survey of machine and deep learning methods for internet of things (iot) security, *IEEE Communications Surveys and Tutorials* **22**(3): 1646–1685.

Aldhaheri, A., Alwahedi, F., Ferrag, M. A. and Battah, A. (2024). Deep learning for cyber threat detection in iot networks: A review, *Internet of Things and Cyber-Physical Systems* **4**: 110–128.

Alwahedi, F., Aldhaheri, A., Ferrag, M. A., Battah, A. and Tihanyi, N. (2024). Machine learning techniques for iot security: Current research and future vision with generative ai and large language models, *Internet of Things and Cyber-Physical Systems* **4**: 167–185.

Aslan, Ö., Aktuğ, S. S., Ozkan-Okay, M., Yilmaz, A. A. and Akin, E. (2023). A comprehensive review of cyber security vulnerabilities, threats, attacks, and solutions, *Electronics* **12**(6): 1333.

Aversano, L., Bernardi, M. L., Cimitile, M. and Pecori, R. (2021). A systematic review on deep learning approaches for iot security, *Computer Science Review* **40**: 100389.

Chander, B. and Gopalakrishnan, K. (2019). Security vulnerabilities and issues of traditional wireless sensors networks in iot, *Intelligent Systems Reference Library*, Vol. 174, Springer, Cham, pp. 519–549.

Chernis, B. and Verma, R. (2018). Machine learning methods for software vulnerability detection, *IWSPA 2018 - Proceedings of the 4th ACM International Workshop on Security and Privacy Analytics*, pp. 31–39.

Dhirani, L. L., Armstrong, E. and Newe, T. (2021). Industrial iot, cyber threats, and standards landscape: Evaluation and roadmap, *Sensors* **21**(11): 3901.

Gupta, A., Anpalagan, A., Carvalho, G. H. S., Khwaja, A. S., Guan, L. and Woungang, I. (2019). Prevailing and emerging cyber threats and security practices in iot-enabled smart grids: A survey, *Journal of Network and Computer Applications* **132**: 118–148.

Hanif, H., Nasir, M. H. N. M., Ab Razak, M. F., Firdaus, A. and Anuar, N. B. (2021). The rise of software vulnerability: Taxonomy of software vulnerabilities detection and machine learning approaches, *Journal of Network and Computer Applications* **179**: 103009.

Khan, A. R., Kashif, M., Jhaveri, R. H., Raut, R., Saba, T. and Bahaj, S. A. (2022). Deep learning for intrusion detection and security of internet of things (iot): Current analysis, challenges, and possible solutions, *Security and Communication Networks* **2022**(1): 4016073.

Moustafa, N. and Slay, J. (2015). UNSW-NB15: A Comprehensive Data Set for Network Intrusion Detection Systems (UNSW-NB15 Network Data Set), *Military Communications and Information Systems Conference (MilCIS)*, IEEE.

Naveen and Sharma, U. (2022). Reduction of iot security vulnerabilities using machine learning algorithm, *Lecture Notes in Electrical Engineering*, Vol. 915, Springer, pp. 677–687.

Pistoia, M., Chandra, S., Fink, S. J. and Yahav, E. (2017). A survey of static analysis methods for identifying security vulnerabilities in software systems, *IBM Systems Journal* **46**(2): 265–288.

Pourrahmani, H., Yavarinasab, A., Monazzah, A. M. H. and Van herle, J. (2023). A review of the security vulnerabilities and countermeasures in the internet of things solutions: A bright future for the blockchain, *Internet of Things* **23**: 100888.

Shahriar, H. and Zulkernine, M. (2022). Mitigating program security vulnerabilities: Approaches and challenges, *ACM Computing Surveys* **44**(3).

Shokry, M., Awad, A. I., Abd-Ellah, M. K. and Khalaf, A. A. M. (2022). Systematic survey of advanced metering infrastructure security: Vulnerabilities, attacks, countermeasures, and future vision, *Future Generation Computer Systems* **136**: 358–377.

Ullah, F., Naeem, H., Jabbar, S., Khalid, S., Latif, M. A., Al-Turjman, F. and Mostarda, L. (2019). Cyber security threats detection in internet of things using deep learning approach, *IEEE Access* **7**: 124379–124389.

Waqas, M., Kumar, K., Laghari, A. A., Saeed, U., Rind, M. M., Shaikh, A. A., Hussain, F., Rai, A. and Qazi, A. Q. (2022). Botnet attack detection in internet of things devices over cloud environment via machine learning, *Concurrency and Computation: Practice and Experience* **34**(4): e6662.