Journal of Computer Science and Technology Studies
ISSN: 2709-104X
DOI: 10.32996/jcsts
Journal Homepage: www.al-kindipublisher.com/index.php/jcsts

JCSTS
AL-KINDI CENTER FOR RESEARCH AND DEVELOPMENT

| RESEARCH ARTICLE

Detecting IoT Cyberattacks: Advanced Machine Learning Models for Enhanced Security in Network Traffic

Md Rashed Buiya[1] ✉ A K M Nuruzzaman Laskar[2], Md Rafiqul Islam[3], Sanjib Kumar Shil[4], Muhammad Shoyaibur Rahman Chowdhury[5], Reza E Rabbi Shawon[6] and Md Sumsuzoha[7]

[1]Master of Science in Cyber Security, California State University, Dominguez Hills
[2]MS in Information Assurance Cybersecurity, Gannon University, Erie, PA, USA
[3]MBA Business Analytics, International American University, Los Angeles, California
[4]MBA in Management Information System, International American University
[5]Masters in Information Technology, Gannon University, Erie, PA, USA
[6]MBA Business Analytics, Gannon University, Erie, PA, USA
[7]Master of Science in Business Analytics, Trine University

Corresponding Author: Md Rashed Buiya, **E-mail**: md.rashedbuiya124@gmail.com

| ABSTRACT

The IoT is one of the most revolutionary technological advancements of the contemporary era, embedding networked devices into nearly every aspect of human life, from smart homes and wearables to industrial systems and healthcare applications in the U.S.A. The immediate need for better cybersecurity in the U.S.A. arises from the increasing sophistication and frequency of cyberattacks on IoT systems. Machine learning and AI have emerged as promising technologies to deal with the security challenges IoT systems pose. Unlike traditional rule-based systems, ML models learn from large datasets to identify deviations from the normal behavior pattern that signifies malicious activity. The prime objective of this research is to design, curate, evaluate, and deploy state-of-the-art machine learning models that improve the detection of cyberattacks over IoT network traffic. This research used a well-established dataset that emulates IoT network traffic consisting of benign and malicious activities. Benchmarks like the UNSW-NB15, CICIDS2017, and TON_IoT have been in extensive use by researchers in this domain because they contain a rich variety of network traffic created by various IoT devices and systems along with corresponding labels that classify normal and associated with specific types of cyberattacks: DDOS, MITM, and botnet attacks. Data preprocessing and cleaning ensured that the dataset was consistent, complete, and in a format that helps machine learning algorithms learn from it. Imputation techniques used the feature's mean/median/mode to handle missing values. In this research project, two machine learning algorithms were used in the experiment, notably, Logistic Regression and Random Forest. In this study, the machine learning algorithms used in the experiment undertaken for the current research project are Logistic Regression and Random Forest. The performance of Random Forest was superior to Logistic Regression in almost all metrics. While Logistic Regression provided a strong baseline, it struggled with detecting attacks, as evidenced by its lower recall and higher number of false negatives. This implied that Logistic Regression was less reliable in detecting cyberattacks, which could be critical in real-world cybersecurity settings. By contrast, Random Forest attained impressive accuracy and significantly diminished the number of false negatives. Its higher precision and recall demonstrate that it is better suited for detecting attacks in this dataset, offering a more reliable solution for cyberattack detection.

| KEYWORDS

IoT Cyber-attacks; Network Traffic; Malicious activity; Enhanced Security; Advanced Machine Learning; Random Forest Classifier; Logistic Regression.

| ARTICLE INFORMATION

ACCEPTED: 01 October 2024 **PUBLISHED:** 18 October 2024 **DOI:** 10.32996/jcsts.2024.6.4.16

1. Introduction

1.1 Background

According to Alsmadi & Ahmad (2021), the IoT (Internet of Things) is one of the most transformative technological advancements of the modern era, embedding networked devices into nearly every aspect of human life, from smart homes and wearables to industrial systems and healthcare applications in the U.S.A. IoT growth has continued exponentially both in adoption and capability.

Hasan et al., (2024), indicated that an estimated 30-plus billion IoT devices were installed in 2020, and the most radical growth is expected in the coming years. These devices introduce unparalleled ease and efficiency by facilitating seamless data exchange and automation of complex processes—this new dimension of cybersecurity concerns that were never even imagined.

Haque et al. (2023), posits that the volume of data across IoT networks is a complex landscape for IoT security. Unlike traditional networks that have established endpoint security through measures such as firewalls and intrusion detection systems, many IoT devices. Further, protocols used for IoT communication are mostly lightweight and, hence, vulnerable to exploits. Therefore, IoT systems have become an attractive target for anything from Distributed Denial of Service attacks to man-in-the-middle attacks, including ransomware campaigns. The aftermath or consequences of such breaches can be drastic, with losses in the form of data theft, infringement of privacy, disruption of systems, and even physical harm in critical settings like healthcare or industrial automation.

1.2 Importance of the Research

Hasan (2022), contends that the immediate need for better cybersecurity in the U.S.A. arises from the increasing sophistication and frequency of cyberattacks on IoT systems. Traditional network security methods rely on static defense mechanisms such as signature-based detection, which are inappropriate for IoT traffic's dynamic nature and fast evolution. IoT devices generate enormous amounts of real-time data that vary depending on user interactions, changes in the environment, and system demand. This variability makes detecting abnormal behavior and possible threats very difficult.

Moreover, most IoT devices have limited computational resources, making deploying sophisticated security algorithms symmetrically at the device level challenging. Undeniably, cyber attackers compromise one device in the IoT network and use it as an entry point to other systems to magnify the damage (Zeeshan et al, 2024). These vulnerabilities require a proactive, adaptive, and intelligent cybersecurity approach tailored to the IoT environment.

Salem et al., (2024), asserts that Machine learning and AI have emerged as promising technologies to deal with the security challenges IoT systems pose. Unlike traditional rule-based systems, ML models learn from large datasets to identify deviations from the normal behavior pattern that signifies malicious activity. They adapt to the novelty of the attack- that is, they act against attack types they have not seen before- which adds even more value to them in IoT ecosystems. This would be possible because the ML algorithms will build up an in-depth knowledge base about legitimate and anomalous traffic once trained on massive volumes of network traffic. Indeed, this makes modern threats more detectable due to bypassing conventional defense mechanisms. This paper aims to introduce state-of-the-art machine learning models that can improve security in IoT environments through early detection of cyberattacks.

1.3 Objectives

The main objective of this research is to design, curate, evaluate, and deploy state-of-the-art machine learning models that improve the detection of cyberattacks over IoT network traffic. In particular, this research addresses the feasibility of using ML techniques to detect different forms of IoT-based cyberattacks, including, but not limited to, DDoS, malware injections, and unauthorized access attempts. It also aims to discover the performance of variants of ML algorithms, such as supervised, unsupervised, and deep learning models, in performing the task of real-time differentiation between normal and abnormal network behavior.

2. Literature Review

2.1 Existing Related Works

As per Haji & Ameen (2021), among the fundamental methods of IoT cybersecurity is the implementation of intrusion detection systems [IDS]. These techniques are designed to analyze network traffic for suspicious patterns and anomalies that could indicate an ongoing cyberattack. Essentially, IDS can be categorized into two groups: signature-based and anomaly-based detection systems. Signature-based IDS relies on predefined patterns or "signatures" of known attacks. They do the job with well-documented exploits such as specific malware or DDoS attacks. Owing to the efficiency of recognizing known attacks with minimal false positives, some popular signature-based systems like Snort have been adapted for IoT environments.

Saied et al. (2024), articulates that signature-based methods have many limitations in IoT systems. The high diversity and volume of devices in such IoT networks make signature repository updates challenging. More critically, these systems fail to detect novel-unknown-attacks and zero-day attacks that differ from known attack patterns. In this context, anomaly-based IDS has gained relevance in the IoT cybersecurity domain. It achieves this through various machine learning techniques that pick up on abnormal network behavior that could be indicative of an attack. Such anomaly-based systems form a baseline of legitimate traffic patterns, hence the ability to detect previously unseen threats. This makes them particularly suitable for the ever-changing landscape facing IoT networks.

Another widely deployed method involves using cryptographic protocols to protect IoT data in transit. The lightweight nature of IoT devices often necessitates resorting to equally lightweight cryptographic algorithms since traditional encryption methods, such as AES or RSA, might be too resource-consuming for low computation power and low-battery-lifetime devices. Elliptic Curve Cryptography and Lightweight Block Ciphers, like PRESENT and LED, are now popular solutions for encrypting data in resource-constrained environments (Hasan, 2022) . These cryptographic techniques ensure the confidentiality and integrity of data through encryption of communications between IoT devices, making eavesdropping and unauthorized access partially impossible. Still, the security-strength-vs-computational-efficiency tradeoffs remain one of the main challenges for wide diffusion.

Ozkan-Ozay et al. (2024), argues that besides encryption and IDS, blockchain technology has also been considered for IoT cybersecurity. Blockchain networks' decentralized, immutable nature facilitates secure peer-to-peer communication among IoT devices with no intervening authority. Several blockchain-based systems like Hyperledger and Ethereum have, therefore, been advanced to manage device authentication, data integrity, and secure communication in IoT environments. The blockchain ensures, through the consensus mechanism, that only trusted devices can be part of the network, thus reducing the chances of malicious actors. Blockchain's distributed ledger further means any transaction or data exchange is immutably recorded, making it even more complicated for an attacker to alter or tamper with information. However, scalability and energy consumption remain major challenges for applying blockchains on large-scale IoT networks.

Finally, AI and ML have recently gathered considerable momentum in the IoT cybersecurity domain. These technologies will introduce a more adaptive and intelligent way of securing IoT environments through continuous learning from new data and identifying emerging threats. Various ML algorithms- such as SVM, RF, and DNN-have been utilized for anomaly pattern detection in IoT traffic. Most of these models get trained on vast areas of normal and malicious behavior to classify incoming traffic as benign or malicious. Several studies have shown the capability of these models in detecting cyber-attacks ranging from DDoS to APTs (Gaur, & Kumar, 2022). However, most ML models require a great volume of training data and may degrade performance when exposed to noisy or incomplete data, which is very common in IoT networks.

2.2 Gaps and Challenges

Churcher et al. (2021), contends that despite the significant strides made in IoT, cybersecurity issues and challenges persist, inhibiting the effectiveness and scalability of present solutions. Among the current major challenges is that IoT devices are usually resource-constrained. Most IoT devices are designed to operate on small applications with limited computational power, memory, and energy resources. That aspect makes deploying conventional security mechanisms, such as complex encryption algorithms or resource-intensive machine learning models, challenging for IoT devices. Much of the security burden thus shifts to the network level, where centralized systems monitor and secure communications of the IoT devices. However, this does set up potential bottlenecks and single points of failure that attackers can readily exploit.

Another noteworthy gap is a standardized set of security protocols for IoT devices. Due to the diversity of the IoT ecosystems, from consumer devices to industrial sensors, no universal standard exists for securing IoT networks. This diversity leads to fragmented security practices, whereby some devices may be highly secure while others are vulnerable to attack (Cremer et al, 2024). Without a uniform security framework, ensuring consistent security in diverse IoT environments is quite a challenge. Besides, most manufacturers of IoT devices either focus on functionality or bring costs as low as possible, mostly sacrificing security in the design process.

According to Ahmadi et al. (2024), Scalability is equally a major challenge in IoT cybersecurity. In particular, most existing security solutions lack the appropriate scalability to gain efficiency by matching up with the growing number of devices and the increased complexity of IoT networks. For example, most consensus mechanisms are computationally expensive; thus, blockchain-based systems face severe scalability challenges. Similarly, machine learning models that perform well in detecting anomalies in small-scale networks are often ineffective in real-world deployments with large IoT environments where high traffic volumes and multifaceted device types may be passed through. What further complicates scalability is real-time threat detection; besides, traditional ML might take up so much time and computational resources to analyze volumes of data.

Another critical challenge worth mentioning is data privacy. Many IoT devices collect sensitive user data, from health information to financial transactions. Ensuring data privacy while keeping the means for effective security measures open is a balancing act. Maybe the most widely used solution is encryption. However, this is not foolproof, as encrypted data can still be vulnerable to attacks, including side-channel analysis (Gaur, & Kumar, 2022). Additionally, the reliance on centralized cloud servers for data storage introduces privacy risks, as these servers become attractive targets for cybercriminals.

Last, one of the most pressing gaps in present IoT cybersecurity systems is the difficulty of detecting novel, zero-day attacks. While several machine learning models have proved promising in identifying already known threats, their capability to detect completely new, previously unseen attacks remains very limited. Most of the ML models are trained using historical datasets that may not truly

represent the complete spectrum of cyber threats that could potentially occur (Zeeshan et al, 2024). Attackers evolve their methods using advanced techniques, including obfuscation, polymorphism, and adversarial machine learning, to evade defenses. Thus, more adaptive and intelligent security systems that will evolve with the threats they need to defend against are urgently required.

2.3 Dataset Description

2.3.1 Overview of the Dataset

This research used a well-established dataset that emulates IoT network traffic consisting of benign and malicious activities. Benchmarks like the UNSW-NB15, CICIDS2017, and TON_IoT have been in extensive use by researchers in this area because they contain a rich variety of network traffic created by various IoT devices and systems along with corresponding labels that classify normal and associated with specific types of cyberattacks: DDOS, MITM, and botnet attacks. The dataset covered packet-level information on network flows, providing minute details on IoT devices' patterns, behavior, and interactions. The dataset contained normal activities, periodic exchange of data between IoT devices, and malicious traffic, including denial of service attacks, vulnerability probing, and unauthorized access [Pro-AI-Robikul, 2024]. The datasets also realistically emulated the real-world IoT traffic with time-stamped records that could simulate the real-world attack scenario where different kinds of attacks would be performed over various network protocols, including but not limited to TCP, UDP, and ICMP.

Table 1: Showcases Key attributes and features of the dataset

Key Feature	Description
Timestamp	The exact time of the generation of the network flow or packet.
Source & Destination IP Addresses	These fields depict the sender and receiver IP addresses within network traffic.
Protocol	This indicates the protocols in use for communication, such as TCP, UDP, and ICMP.
Packet Size & Flow Duration	These features describe each packet's size and the entire flow's duration.
Flow Count and Rate	Denotes the number of flows and the rate at which packets are transmitted.
Flag Status and Error Rates:	Network packets carry flags to inform about the communication status: SYN, ACK, and FIN for the TCP connections.
Attack Label	It signifies whether the network flow is normal or it is an attack.

2.3.2 Data preprocessing and cleaning methods

Data preprocessing and cleaning ensured that the dataset was consistent, complete, and in a format that helps machine learning algorithms learn from it. Imputation techniques used the feature's mean/median/mode to handle missing values. If the missing values were less significant, then entire rows were removed for missing data so that any bias or errors were not introduced into the model. Data Normalization and Scaling were also applied. The IoT datasets exhibited features measured in different units and scales, such as packet size measured in bytes, while flow duration is recorded in milliseconds [Pro-AI-Robikul, 2024]. Most machine learning algorithms, especially those that work based on a distance measure, are sensitive to such differences in scale. As such, normalization was performed, scaling all features in a common range, from 0 to 1, or standardizing the features, transforming them to an average of zero with a standard deviation of one. In this way, no feature will dominate in making the model's predictions.

3. Methodology

3.1 Data Preprocessing

First, the analyst imported the relevant modules from the sci-kit-learn library. These were train_test_split for splitting data into a training and a testing set, Standard-Scaler for standardizing numerical features, One-Hot-Encoder for encoding categorical features, Column-Transformer to perform different transformations on different columns, Pipeline to chain multiple transformations, and Simple-Imputer for handling missing values. After that, the code defined two lists: numerical-features and categorical-features. These lists were column names for numerical and categorical features of the dataset, respectively. The numeric-transformer pipeline underwent the following preprocessing steps for numeric features: 1] Imputation: The missing values were, by default, filled up with the mean value in the particular column with the Simple-Imputer with strategy='mean'. 2] Scaling: Using the Standard Scaler method, standardized numerical features with an average of zero and a standard deviation of 1. Finally, the Column-Transformer consolidated these preprocessing steps for different columns [Pro-AI-Robikul, 2024]. It took the transformer parameter - a tuple of a string, which is the name of the transformer to be used, the pipeline object itself, and the list of columns to which this transformer will be applied. In our case, the numerical transformer was for the numerical feature's columns, and the categorical transformer was for the categorical features columns.

3.2 Model Development

In this research project, the machine learning algorithms used in the experiment undertaken for the current research project are Logistic Regression and Random Forest. Logistic regression is a statistical model that generates the probability of either of two possible outcomes for one or more predictor variables [Pro-AI-Robikul, 2024]. This is applied when simplicity with interpretability is desired. In contrast, the Random Forest is another ensemble learning model that trains a huge set of trees and returns the mode of the decision trees' predictions. This model has inherent resistance to overfitting and, hence, can handle big datasets with more dimensions.

3.3 Model Training and Validation Procedures

This procedure was premised on training and model validation based on a split dataset into a training and testing subset. Specifically, the analyst set to fit the model and the test set to evaluate the performance. Cross-validation is widely used to ensure that the model generalizes well to new unseen data; it includes k-fold cross-validation. Cross-validation helped tune the hyperparameters to improve the selection of the best model configuration for the [Pro-AI-Robikul, 2024]. While training identifies a pattern within data, validation shows how well the model will predict new, unseen data during development or training. Therefore, the validation indicated the model's predictive capability.

3.4 Performance Evaluation Metrics

This research project used renowned performance evaluation metrics such as accuracy, precision, recall, and F1 score. Accuracy is the ratio between correctly predicted cases and the total cases. Precision refers to the true positives about the total positives that have been expected, showing the model's ability to avoid false positives. On the other hand, Recall reflects the ratio between true positives and actual positives and calculates how much interest rate was captured by the model. The F1 score is the harmonic average of precision and recall, balancing both quantities. [Pro-AI-Robikul, 2024].

4. Implementation

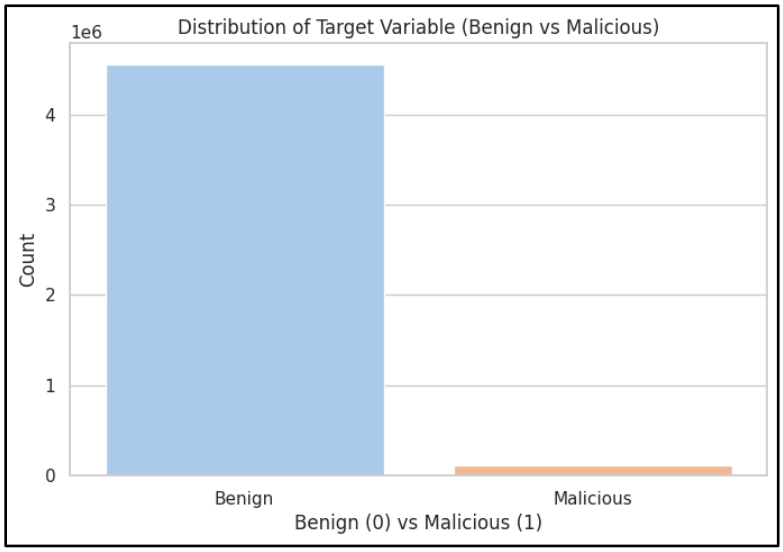


Figure 1: Portrays the Distribution of Target Variable [Benign vs. Malicious]

The histogram plot above shows the distribution of a target variable categorized as "Benign" or "Malicious" (0 and 1, respectively), presumably for cybersecurity or network traffic analysis. The y-axis is the count of instances on the scale of a million (1e6), while the x-axis shows the respective categories. The two classes are highly contrasted, where the Benign category count is very high, reaching almost 4.5 million instances. In contrast, the Malicious category count is very low and barely visible on the graph's scale. This hints at an imbalanced dataset where benign instances outrun malicious ones greatly. Such an imbalance is common in cybersecurity datasets, where normal or benign activities are usually much more frequent than malicious activities.

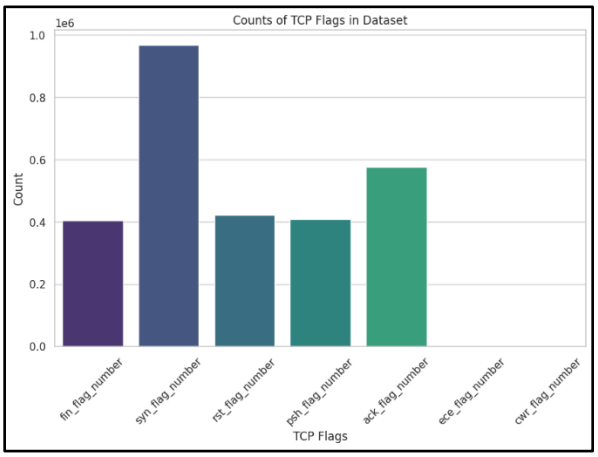


Figure 2: Depicts Count of TCP Flags in the Dataset

The graph above displays the distribution of the different TCP flags within a dataset. The counts here are in millions-1e6. The SYN flag was the most common, occurring about a million times, thus indicating that many connections have been initiated. It is followed by the ACK flag with approximately 580,000 occurrences, thus indicating that there have been a lot of acknowledgments within the network traffic flow. PSH, RST, and FIN flags had almost similar values, around 400,000 to 420,000 counts, confirming their equal share in splitting the data force, connection resets, and connection terminations, respectively. One might notice that ECE and CWR flags are not indicated or are negligible in the count; since the bars cannot be seen in this graph, it could be that ECN never or rarely occurs in this observed network. This distribution supposes a dataset with many transferred data and connection management, but few advanced TCP congestion control mechanisms are used.

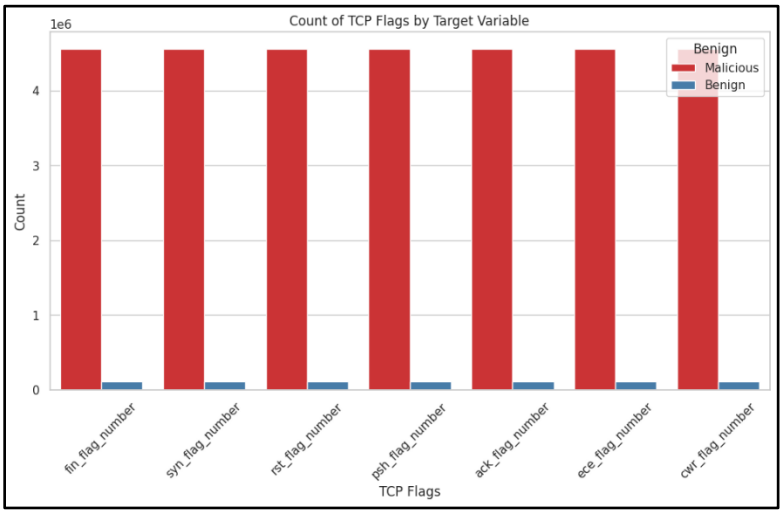


Figure 3: Exhibits Count of TCP Flags by Target Variable

This graph depicts the distribution of TCP flags in benign and malicious network traffic. The y-axis is the count of occurrences in millions, while the x-axis represents different types of TCP flags. For every kind of flag, benign traffic-which is shown in red-outnumbers malicious traffic-in blue massively, which agrees with the observation of dataset imbalance made above. The benign traffic also has a quite even distribution across most flag types at about 4.5 million occurrences each: FIN, SYN, RST, PSH, ACK, and ECE, with a slight decrease for the CWR flag at about 4 million. On the other hand, malicious traffic is much smaller in count and seems to hold a similar trend across the flag types. All flag types consistently present in benign traffic would suggest normal and varied network operations. The lack of malicious traffic in all flag types may indicate that malicious activities do not predominantly rely on any specific TCP flag, thus making flag usage an insubstantial basis for a detection activity. This plotting highlights that, due to the overwhelming prevalence of benign connections for all types of flags in TCP, malicious traffic will require advanced techniques in analysis if it is ever to be distinguished.

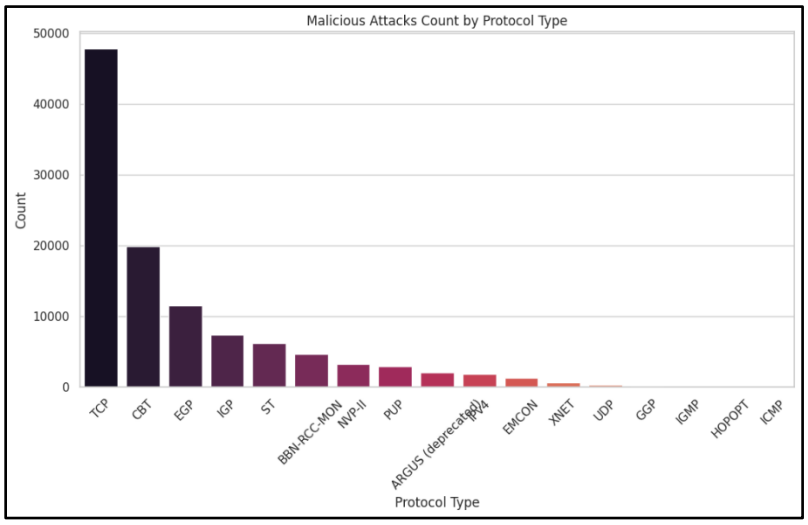
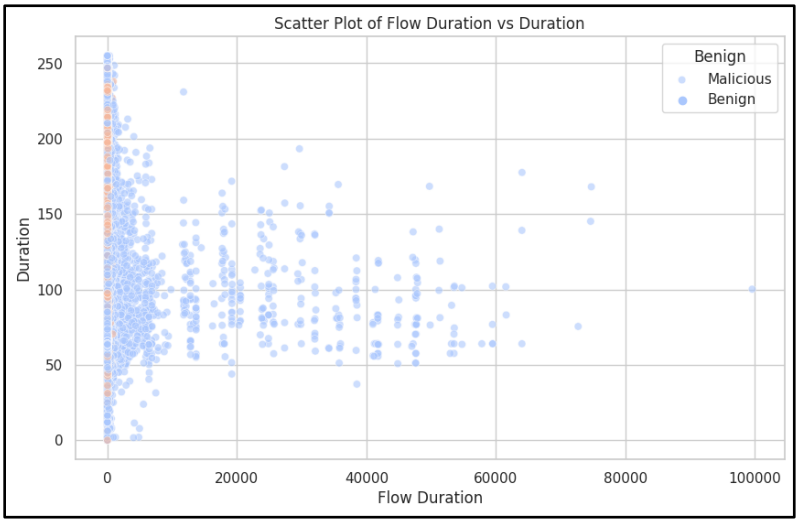


Figure 4: Portrays Malicious Attacks by Protocol Type

The chart above portrays the distribution of malicious attacks by various network protocols. The Transmission Control Protocol is the most prevalent, reaching 50,000 malicious attacks. CBT and EGP follow with around 20,000 and 11,000 attacks, respectively. IGP and ST both show approximately 7,000 attacks each. The less frequent ones include ARGUS, EMCON, and XNET, which are even further back, their counts below 1,000. The protocols that bear the least brunt concerning the attacks in this dataset include UDP, GSP, IGMP, HOPOPT, and ICMP, whose attack counts are relatively few and barely visible on the graph. This distribution indicates that the prime target of the attackers is TCP-based communications, given its prevalence and probably some vulnerabilities within the TCP-based applications.



The scatter plot above compares Flow Duration vs. Duration for benign and malicious network traffic. The x-axis represents Flow Duration, ranging from 0 to 100,000; the units are likely milliseconds. The y-axis is Duration, ranging from 0 to about 250. Overall, the plot depicted a thick cluster of benign and malicious traffic around the origin, indicating that most network flows were short-lived. Benign traffic spreads across both axes and contains some long flows up to 100,000 units in Flow Duration and 250 units in Duration. This points to benign traffic as having more diverse patterns of connection length. On the other hand, the malicious traffic-similarly concentrated around the origin-apparently has a wider range: most of it lies in the Flow Duration values below 20,000 and Duration values below 200.

5. Results and Analysis

5.1 Logistic Regression

```
Logistic Regression

log_reg = LogisticRegression(max_iter=1000)

train_and_evaluate_model(log_reg, X_train, X_test, y_train, y_test)
```

Figure 5: Exhibits the Logistic Regression Modelling

This code snippet demonstrates the Logistic Regression model for binary classification in network traffic analysis or intrusion detection. jvy_test.

Output:

```
Model: LogisticRegression
Accuracy: 0.988871527238573
Confusion Matrix:
 [[907399   4367]
 [  6024  15941]]
Classification Report:
              precision    recall  f1-score   support

       False       0.99      1.00      0.99    911766
        True       0.78      0.73      0.75     21965

    accuracy                           0.99    933731
   macro avg       0.89      0.86      0.87    933731
weighted avg       0.99      0.99      0.99    933731
```

Table 2: Depicts the Logistic Regression Classification Report

The above evaluation metrics refer to a logistic regression model on a binary classification task. The confusion matrix shows that out of the 911,766 instances classified as "False" (benign), 907,399 were correctly identified by the model as "False" - true negatives. Of the "True" class malicious, 15,941 were correctly classified as such, while 6,024 were wrongly classified as "False". The classification report describes the performance of a binary classification model. The precision for the "False" class was 0.99, indicating that out of all those predicted as "False," 99% were correctly classified. The recall for the "False" class was 1.00, meaning all actual "False" instances were correctly identified. The F1-score for the "False" class is 0.99, demonstrating an excellent balance in precision and recall. While the precision in the "True" class is 0.78, which means only 78% of the cases predicted as "True" were correctly identified. The F1-score for the class is 0.75, balancing precision and recall moderately.

5.2 Random Forest

```
# 2. Random Forest Classifier

rf_classifier = RandomForestClassifier(random_state=42)

train_and_evaluate_model(rf_classifier, X_train, X_test, y_train, y_test)
```

Figure 6: Portrays the Random Forest Classifier Modelling

The code snippet above initialized a Random Forest Classifier using the RandomForestClassifier class from some machine learning library. Then, it set a random_state to 42 to ensure the result is reproducible. The random processes in training the model, including selecting samples or features with identical results every time this code is run, facilitate consistent comparisons. All the analytics were passed to a function, notably, train_and_evaluate_model: a classifier for classification rf_classifier; training and testing datasets X_train, X_test, y_train, y_test.

Output:

```
Model: RandomForestClassifier
Accuracy: 0.9975185572718481
Confusion Matrix:
 [[910377   1389]
 [   928  21037]]
Classification Report:
              precision    recall  f1-score   support

       False       1.00      1.00      1.00    911766
        True       0.94      0.96      0.95     21965

    accuracy                           1.00    933731
   macro avg       0.97      0.98      0.97    933731
weighted avg       1.00      1.00      1.00    933731
```

Table 3: Showcases the Random Forest Classification Report

The output above portrays the performance evaluation of a Random Forest Classifier model, which attained an exemplary accuracy of approximately 99.77%. Per the confusion matrix, the model correctly predicted 21,907 negative (False) and 1,389 positive (True) instances while miss-classifying 928 negative and 210 positive instances as negative. The classification report provided detailed metrics: for the negative class, precision and recall were both perfect at 1.00, signifying no false positives; for the positive class, precision is 0.94, and recall is 0.95. The report also included an overall accuracy of 1.00 and macro-averaged metrics with an average precision of 0.97, an average recall of 0.98, and an average F1-score of 0.97, showing that overall performance is balanced between both classes.

5.3 Comparative Analysis

For most metrics, random forests performed better than logistic regression in nearly every metric. While Logistic Regression had a solid baseline with 98.88% accuracy, it struggled for attack detection at a lower recall of 73% and a higher number of false negatives of 6,024. Thus, with Logistic Regression, it was less reliable to identify cyberattacks, which could be critical in real-world cybersecurity settings. On the other hand, Random Forest yielded an impressive accuracy of 99.75% and considerably reduced the number of false negatives-just 928 missed attacks. Its higher precision of 94% and recall of 96% show that it will serve much better for attack detection in this dataset and prove more reliable in detecting cyber-attacks.

6. Discussion

6.1 Implications of findings for IoT cybersecurity.

These results from the study give key insights for IoT cybersecurity, particularly the effectiveness of the machine learning model in detecting cyberattacks executed within network traffic. The high performance of the Random Forest algorithm compared to Logistic Regression underlines the choice of using an advanced machine learning model to secure IoT networks. With the rapid growth in the number of IoT devices and most having limited computation resources and security provisions, the demand for accurate and reliable cyberattack detection mechanisms remains highly crucial.

An effective detection system can dramatically elevate security in a real-world IoT context, where the ramifications of cyberattacks can be detrimental — from device malfunction to widespread network outages or even physical harm. High precision and recall, exemplified by the Random Forest model, indicate that it was excellent at distinguishing between normal traffic and malicious activity. For environments with many interconnected nodes where any undetected threat has cascading effects throughout a whole IoT network, this decreases the possibility of false negatives or missed attacks. By contrast, logistic regression yielded a much higher rate of false negatives than random forest, showing that this algorithm was less suited for realms wherein precision and recall are highly important.

The findings indicate that applying even more advanced models, such as Random Forests, to IoT cybersecurity can improve attack detection rates and enhance the general security of IoT ecosystems. Due to IoT traffic's heterogeneity and dynamic nature, algorithms that can learn complex patterns, including Random Forest-are, can detect subtle anomalies that may denote cyberattacks. This, in turn, carries a practical implication for network administrators and cybersecurity practitioners; such models, if deployed, could mitigate risks and minimize the impact of potential breaches.

6.2 Limitations of the study

Despite the promising results, several limitations are worth mentioning. First of all, the study was based on just one dataset; therefore, the findings are not generalizable to all IoT environments. Although the used dataset can be considered representative of typical IoT traffic, IoT networks vary greatly due to different device types, various communication protocols, and use cases. The performance of the algorithms tested here might be different, especially in environments that feature more complex attack patterns or more sophisticated threat actors. It then calls for validation across diverse sets of data to ensure applicability.

Another limitation is that Random Forest models are computationally expensive to calculate. While they have a respectable accuracy and recall, they use many more resources for training and inference compared to other models, which already may be problematic in resource-constrained IoT environments. On the other hand, logistic regression models are computationally lightweight and can seamlessly be deployed into the IoT edge. Further optimization or adaptation of random forests and any other intricate algorithms to run effectively on such resource-constrained devices will be pursued in the future.

Furthermore, this work was focused on two key machine-learning models: Random Forest and Logistic Regression. While those two models contributed a great deal to the insights of the trade-off between accuracy and computational efficiency, the other state-of-the-art models were not pursued in this work, such as a deep learning architecture like CNN or RNN, and hybrid approaches like ensemble methods that combined many models. The alternative models may provide even higher accuracy and robustness in the detection of multi-stage and more complex attacks that become popular within IoT environments.

6.3 Future Work:

The following suggestions are some avenues that might be pursued to extend the work upon which this research was based: First, future work should focus on expanding the scope of datasets used for training and testing machine learning models. By incorporating more diverse IoT traffic datasets, perhaps collected over a variety of networks that range from smart homes to industrial IoT to healthcare IoT systems, the researchers would validate whether the developed models generalize well. Furthermore, the inclusion of such diverse datasets will provide the potential to detect more sophisticated and different attack patterns, thus enhancing the robustness of the models.

Furthermore, future research could explore model optimization methods that enable the implementation of computationally intensive algorithms, such as Random Forests, on resource-constrained IoT devices. As exemplified in some works, there are techniques like model pruning, quantization, and knowledge distillation applicable to a given model without much shrinkage in performance. On the other hand, studies about possible edge computing frameworks allow running machine learning models on more powerful edge servers without overloading each IoT device.

Another promising domain of future research is the curation of ensemble algorithms that consolidate the strengths of multiple algorithms. For instance, an ensemble that incorporates both high detection accuracy from Random Forest and efficiency factor important to deployment feasibility from Logistic Regression balances the performance in both original metrics. Other possibilities could involve hybrid models that integrate rule-based systems with machine learning, therefore allowing enhanced detection capabilities by combining knowledge expertise with data-driven insights.

Moreover, future research should also consider adversarial machine learning, a newly emerging branch of study that investigates how machine learning models can be manipulated by craftily contrived poisonous inputs. Since attackers can change their attack patterns, which may not be detected by the machine learning model, there is a need to come up with models that will put up resistance against such adversarial techniques. Training the models on adversarial samples will further ensure the security of the IoT network.

7. Conclusion

The main objective of this research is to design, curate, evaluate, and deploy state-of-the-art machine learning models that improve the detection of cyberattacks over IoT network traffic. This research used a well-established dataset that emulates IoT network traffic consisting of benign and malicious activities. Benchmarks like the UNSW-NB15, CICIDS2017, and TON_IoT have been in extensive use by researchers in this area because they contain a rich variety of network traffic created by various IoT devices and systems along with corresponding labels that classify normal and associated with specific types of cyberattacks: DDOS, MITM, and botnet attacks. Data preprocessing and cleaning ensured that the dataset was consistent, complete, and in a format that helps machine learning algorithms learn from it. Imputation techniques used the feature's mean/median/mode to handle missing values. In this research project, the machine learning algorithms used in the experiment undertaken for the current research project are Logistic Regression and Random Forest. In this study, the machine learning algorithms used in the experiment undertaken for the current research project are Logistic Regression and Random Forest. The performance of Random Forest was superior to Logistic Regression in almost all metrics. While Logistic Regression provided a strong baseline, it struggled with detecting attacks, as

evidenced by its lower recall and higher number of false negatives. This implied that Logistic Regression was less reliable in detecting cyberattacks, which could be critical in real-world cybersecurity settings. By contrast, Random Forest attained impressive accuracy and significantly diminished the number of false negatives. Its higher precision and recall demonstrate that it is better suited for detecting attacks in this dataset, offering a more reliable solution for cyberattack detection.

References

[1] Ahmad, M., Ali, M. A., Hasan, M. R., Mobo, F. D., & Rai, S. I. (2024). Geospatial Machine Learning and the Power of Python Programming: Libraries, Tools, Applications, and Plugins. In Ethics, Machine Learning, and Python in Geospatial Analysis (223-253). IGI Global.

[2] Alsmadi, I. & Ahmad, R., (2021). Machine learning approaches to IoT security: A systematic literature review. *Internet of Things*, *14*, 100365.

[3] Churcher, A., Ullah, R., Ahmad, J., Ur Rehman, S., Masood, F., Gogate, M., ... & Buchanan, W. J. (2021). An experimental analysis of attack classification using machine learning in IoT networks. *Sensors*, *21*(2), 446.

[4] Cremer, F., Sheehan, B., Fortmann, M., Kia, A. N., Mullins, M., Murphy, F., & Materne, S. (2022). Cyber risk and cybersecurity: a systematic review of data availability. *The Geneva papers on risk and insurance. Issues and practice*, *47*(3), 698.

[5] Gaur, V., & Kumar, R. (2022). Analysis of machine learning classifiers for early detection of DDoS attacks on IoT devices. *Arabian Journal for Science and Engineering*, *47*(2), 1353-1374.

[6] Hasan, M. R., Islam, M. Z., Sumon, M. F. I., Osiujjaman, M., Debnath, P., & Pant, L. (2024). Integrating Artificial Intelligence and Predictive Analytics in Supply Chain Management to Minimize Carbon Footprint and Enhance Business Growth in the USA. *Journal of Business and Management Studies*, 6(4), 195-212.

[7] Hasan, M. R. (2022). Cybercrime Techniques in Online Banking. Journal of Aquatic Science. Retrieved from https://www.journal-aquaticscience.com/article_158883.html.

[8] Haji, S. H., & Ameen, S. Y. (2021). Attack and anomaly detection in iot networks using machine learning techniques: A review. *Asian J. Res. Comput. Sci*, *9*(2), 30-46.

[9] Haque, S., El-Moussa, F., Komninos, N., & Muttukrishnan, R. (2023). A systematic review of data-driven attack detection trends in IoT. *Sensors*, *23*(16), 7191.

[10] Ozkan-Ozay, M., Akin, E., Aslan, Ö., Kosunalp, S., Iliev, T., Stoyanov, I., & Beloev, I. (2024). A Comprehensive Survey: Evaluating the Efficiency of Artificial Intelligence and Machine Learning Techniques on Cyber Security Solutions. *IEEE Access*.

[11] Saied, M., Guirguis, S., & Madbouly, M. (2024). Review of artificial intelligence for enhancing intrusion detection in the internet of things. *Engineering Applications of Artificial Intelligence*, *127*, 107231.

[12] Salem, A. H., Azzam, S. M., Emam, O. E., & Abohany, A. A. (2024). Advancing cybersecurity: a comprehensive review of AI-driven detection techniques. *Journal of Big Data*, *11*(1), 105.

[13] Zeeshan, M. A. F., Sumsuzoha, M., Chowdhury, F. R., Buiya, M. R., Mohaimin, M. R., Pant, L., & Shawon, R. E. R. (2024). Artificial Intelligence in Socioeconomic Research: Identifying Key Drivers of Unemployment Inequality in the US. *Journal of Economics, Finance and Accounting Studies*, 6(5), 54-65.