Improving Network and IoT Intrusion Detection Through Machine Learning Algorithms

MSc Practicum 2
Master of Science in Cybersecurity

Prasanth Sriramulu Deenadayala Babu
Student ID: 23173459

School of Computing
National College of Ireland

Supervisor: Vikas Sahni

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: Prasanth Sriramulu Deenadayala Babu

Student ID: 23173459

Programme: MSC Cyber Security **Year:** Jan 2024

Module: MSc Practicum part 2

Supervisor: Vikas Sahni

Submission Due Date: 12-12-2024

Project Title: Improving Network and IoT Intrusion Detection Through Machine Learning Algorithms

Word Count: 6559 **Page Count** 21

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: Prasanth Sriramulu Deenadayala Babu

Date: 12-12-2024

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):	

Improving Network and IoT Intrusion Detection Through Machine Learning Algorithms

Prasanth Sriramulu Deenadayala Babu
X23173459

Abstract

Widespread and increased cyberattack against Internet of Things (IoT) are causing enormous range of problem for individual and organizations. The growing need for these services has a possibility to contain anomalies in the IoT data network has emerged as a key challenge. This research evaluates machine learning algorithms for detecting both traditional network intrusion and IoT network logs. Two datasets have been used IoT network log and Network Intrusion log dataset to classify various models on the performance metrics. To deal the issue of class imbalance and scalability nature in network and IoT dataset, Feature selection like Random Forest (RF), Correlation Coefficient and Cross-Validation & Regularization has achieved 99%, by improving this real-time processing combined with effective anomaly detection ensures threats are identified and mitigated quickly. Supervised learning models Decision Tree and KNN model has shown high accuracy of 99%. The findings show the capacity for modifying the machine learning technique to achieve high accuracy to identify labeled malicious in network traffic and IoT logs.

1 Introduction

The Internet of Things (IoT) is making a paradigm shift in various sectors including healthcare, smart homes, transport, or industrial automation meaning that devices can connect and interact seamlessly over networks across multiple domains. The State of IoT Summer 2024 brought out by IoT Analytics that included a total of 171 pages has highlighted that there are 16.6 billion IoT devices connected at the end of 2023 with 15% of the increase over 2022. IoT Analytics forecasts this to move on to 18.8 billion at the end of the year by growing at a rate of 13% (Satyajit Sinha, 2023). IoT devices are highly vulnerable to cyber threats due to limitations in computing power, memory and other resources. Such weaknesses make IoT systems susceptible to multiple threats such distributed denial of service (DDoS), malware, advanced persistent threats (APTs) which in turn translate to data loss, service failure, or even economic disadvantage. (Sasi *et al.*, 2023). Traditional intrusion detection systems (IDS) on the other hand have several shortcomings that have hindered the provision of adequate security in the case of the IoT due to the complexity and rapidly changing environment of these networks, coupled with new types of zero-day or sophisticated attacks. This indicates a dire need for further examination of efficient intrusion detection systems. Machine Learning (ML) offers a promising solution, enabling IDS to analyse vast amounts of network data, detect anomalies in real-time, and adapt to evolving threats. (Khraisat and Alazab, 2021)

Research improve network and IoT intrusion detection with the help of ML algorithms. Particularly, the study seeks challenges around data imbalance, and scalability issues. Also enhance the security, reliability, and efficiency of IoT systems, ensuring their safe integration into critical infrastructures.

1.1 Motivation

The most significant reason for choosing the research topic is that there has been no study done on this field other than in the traditional IT environment to date. To implement advanced supervised learning models, which would not require large unsecured data with many vulnerabilities that can be easily exploited, to detect unknown threats. Many works have been done on IDS systems by using ML techniques and some of them have been achieved with such a good quality that they could replace human observers. So, in this article, we have included comparison of many different algorithms with advanced supervised learning models used for detection of the unknown threats labeled dataset.

1.2 Research Questions

Which is the best machine learning technique that can be used to detect both known and unknown network and IoT device logs?

1.3 Summary of contents

The report consists of sections such as **related** research, which considers the past research papers of IDS by various authors and provides a proposal for further research areas.

The **Methodology** section explains the steps taken in the study to attain the research result with specific details provided in the **design detail** section. In the **implementation**, **evaluation**, and **discussion** section, code and the tools used for the study are presented, followed by the discussion of the outcome of the experiments. Besides, future work has been explored in the conclusion section.

2 Related Work

Several researchers have conducted a study to improve intrusion detection techniques for developing a new idea for growing cyber threats on IoT systems. DDoS, Botnet, Identity theft and Ransomware attacks are most commonly taking place in IoT device. However, there are only a few surveys or information's that focused on ML/DL.

2.1 Literature Review

(Choudhary, Kesswani and Majhi, 2021) The research problem indicates that improving the connectivity of IoT devices requires protecting these networks against intrusions such as DDoS attacks. Study proposes hybrid Intrusion Detection System (IDS), SVM is designed to detect malicious routes and their achieving 98.68% accuracy rate, they findings emphasize on rule-based or single-model classifier intrusion detection systems, which face resource constraints and are ineffective at addressing zero-day attacks.

(Krishnan, Neyaz and Liu, 2021) discuss identifying network intrusion attacks within IoT architectures. Supervised machine learning such as, DT and SVM are implemented in conventional rule-based IDS and streamlined anomaly-based detection models, but they have low accuracy and adaptability to modern attack. From to their perspective, the suggested framework has the potential to improve detection accuracy by leveraging labeled data, thereby improving real-time responsiveness to known attack patterns.

In (Liu *et al.*, 2020) improvement of IoT network effectiveness using IDS, considering their difficulties were limited processing power and the varitey network protocols. This research suggests the artificial deployment of an IoT Network Intrusion Dataset and machine learning algorithms: Logistic Regression, SVM, KNN, Random Forest, and XGBoost, for real-time anomaly detection. In this classification model due to its low computational requirements, their approach demonstrates suitability for GPU-based applications and achieves high accuracy.

(Liu *et al.*, 2022) addresses that traditional intrusion detection systems often fail to predict cyber threats due to data imbalance and limited feature learning capabilities. In their paper using supervised machine learning techniques such as SVM and RF mentioned that, only focuses on single-task method. To solve these limitations, the work suggested a multi-task deep learning framework handling anomaly detection and clustering. The framework evaluates issues related to imbalanced data by using Autoencoders and contrastive learning, to improve accuracy and reduce false alarm rates.

(Rai, Syamala Devi Professor and Guleria, 2015) discuss the inefficiency of creating decision trees for intrusion detection namely split value calculation and its influence on accuracy and performance of IDS models. Previous approaches used C4.5 decision tree algorithm and feature selection to acquire information, but they used expensive computations and biased towards frequent values. The result shows that approach is simpler, faster to train, and more accurate on NSL-KDD data. They can improve by geometric mean split values further.

(Asharf *et al.*, 2020) represents the increasing number of IoT devices, on lack of computational power and the availability of resources making them susceptible to attacks. In addition, standard IDS methods (signature-based and anomaly-based IDS) are often insufficient because they do not provide novel threats. The paper recommends the adoption of real-time anomaly detection that can respond to new threats.

In (Fu *et al.*, 2022) imbalanced data are not effective in modeling complex spatio-temporal patterns in network traffic. The importance shows on real-time and accurate detection to sensitive sectors like finance and defense. The pervious solution was approaches like Navie baye, SVM and CNNs are required to manual feature engineering which is costly and ineffective in the dynamic attack scenarios. DLNID model that tackles the issue, A bidirectional LSTM and data augmentation using ADASYN which makes manual feature extraction.

The system proposed by (Rodríguez *et al.*, 2022) conveys that critical issue of detecting the zero-day cyber-attack in IoT networks. As suggested that, Traditional methods are depending on machine learning and deep learning technique, however they required extensive labeled data and resources. Improving in detection accuracy, particularly at zero-day attack and show their performance with other DL based methods.

(Fenanir and Semchedine, 2023) evaluates IDS security oriented towards the recognition of users' attempts after a breach, however, there are limitations to the availability of traditional systems such as IDS to troubled IoT environments. Scalability and privacy were compromised, however centralized and distributed learning methods provided high accuracy. DNN, CNN, and LSTM aimed at maintaining data privacy through local device training with reporting of global updates. In this model show best result of 99% accuracy also that data security is maintained, and the model remain scalable.

This (Ruzafa-Alcázar *et al.*, 2023) making IDS more efficient in industrial use cases where data privacy is the main issue for decentralized and heterogenous environments. A Federated Learning (FL) approach was used for privacy-preserving IDS but is not effective with non-independent and identically distributed (non-IID) data and in data aggregation where privacy is a concern. Differential Privacy and Fed+ aggregate approach carried out in this research to reduce privacy attacks.

(Moustafa *et al.*, 2023) The lack of explainability and the wrong interpretation of the devices are what it is all about. Furthermore, for the purposes of trust and security that go along with discussing the need for models to detect cyber attacks by recognizing the existence of security problems. The document speaks of Explainable AI (XAI) for the seniors that the Information Detection System (IDS) will also be attacked. The IDS will also detect the various types of attacks on the IoT system using various AI techniques.

(Thaseen and Kumar, 2013) describes facing issues with redundant features and imbalanced datasets by traditional intrusion detection systems are struggling to achieve high accuracy and low false rate. This report evaluated tree-based classifiers like random forest (RT) method and employed features like CFS & CONS and discretization methods to optimize performance. Their best result shows random forest achieved high performance. Implementing ensemble method by combining RT with neural network or gradient boosting can be further improve detection capability.

In (Panigrahi *et al.*, 2021) high class imbalance causes by biased model towards majority class and resulting lower at accuracy on minority class. This paper proposed using decision tree-based intrusion detection using supervised learning by relative random sampling (SRRS) and using features selection for improving performance. From their perspective, implement generative adversarial networks (GANs) to be synthesized minority class that can improve the performance.

(Bhoi *et al.*, 2021) highlights the challenge of selecting optimal supervised classifier for IDS and evaluating multiple datasets. Analyzed 54 models across six groups using NSLKDD, ISCXIDS2012, and CICIDS2017 datasets, identifying J48Consolidated as the most robust. Ensemble techniques by combining supervised model like decision tree will be suggested in improvement at class imbalance.

(Grimaldi, Mahmood and Gidlund, 2019) highlights the problem faced in IoT device face severe interference in shared 2.4 GHz spectrum. Their previous solution was energy sampling and IDI based feature selection has caused long time delays at sensing and hardware. From their view, light neural network are implemented in together with AI technology.

(Elbasiony *et al.*, 2013) traditional intrusion detection, issues with false alarms and incapacity to identify new threats. They showed importance on significance of resloving limitations in order to improving accuracy and operational efficiency. Pervious work contributed by proposing hybrid IDS model combining RF and K-means for anomaly detection. Followed by continuing to employ deep learning for unsupervised anomaly detection in future work.

(Dhanabal and Shantharajah, 2015) evaluates the NSL-KDD dataset for training and testing IDS while identifying effective classification of algorithms. Enhanced IDS performance addressed by the enabling better NSL-KDD dataset, which resolves bias and redundancy concerns in KDD'99. J48 achieved the maximum accuracy (99.8% for normal traffic). Use of ensemble learning techniques, such as Random Forest or Gradient Boosted Trees, to further increase detection accuracy.

(Amira, Hanafi and Hassanien, 2017) shows challenges of selecting network intrusion detection and classification, focusing on improving accuracy for different attack types. An important problem showed in this study was accurate detection reduces false positives and identifies less-represented attacks (e.g., R2L, U2R). Results showed that Naive Bayes excels for low-represented attacks, while BFTree outperforms for high-represented classes. Integrate an ensemble method combining Naive Bayes and BFTree to leverage Naive Bayes for low-represented attacks.

2.2 Summary of Related work

The summary of the review of the relevant research papers is showed in the table below:

S. No	Writer	Methods used	Results	Performance Rate
1.	(Choudhary, Kesswani and Majhi, 2021)	Hybrid IDS integrating SVM and DNN for IoT networks.	Improves detection speed and accuracy, addressing resource constraints and zero-day attacks.	Accuracy: 98.68%; Outperforms traditional single-model approaches.
2.	(Krishnan, Neyaz and Liu, 2021)	Supervised ML (Decision Tree and SVM)	Enhances detection accuracy via labeled data; adapts to modern attack vectors.	Accuracy improvements noted; real-time responsiveness emphasized.
3.	(Liu *et al.*, 2020)	ML algorithms: Logistic Regression, SVM, KNN, Random Forest, XGBoost	Low computational load makes it suitable for IoT; real-time anomaly detection achieved.	High performance; efficient GPU intergrated IoT networks.
4.	(Guleria, 2015)	IDS based model decision tree with improve with split value calculations.	Enhaces model bias; improves training time and real-time detection accuracy.	NSL-KDD dataset: improved accuracy; geometric mean split showed as future work.
5.	(Thaseen and Kumar, 2013)	Supervised tree-based classifiers using NSL-KDD dataset. Random Forest for feature selection and classification.	Tree-based classifiers provide superior accuracy and efficiency. Random Tree showed high accuracy with reduced false alarms.	Random Tree achieved highest accuracy (99.74%) with minimal false alarm rate.
6.	(Elbasiony *et al.*, 2013)	Proposed a hybrid IDS using Random Forests and Weighted k-means clustering.	Combined misuse and anomaly detection for better performance.	Error Rate: 7.27%, Detection Rate: 91.23%, False Positive Rate: 0.54%

Table 1: Summary of research literature review.

2.3 Literature Review Gap

As per number of related works mentioned above, studies encompass about the machine learning technique, but they lack transparency in their decision-making process. The above efforts try to identify the zero-day attacks, but they frequently face limitations by their reliance on computational resources or type of datasets. Improving the network intrusion and IoT detection using supervised learning model to identify known and unknown threats by classifying by labeled dataset is main goal to achieve in this research.

3 Research Methodology

Development is based on improving the network intrusion and IoT device detection using machine learning algorithms by combining different model features and classifiers. The approach taken place by following steps such as preprocessing, analysis of data, implementation of environment and limitations. The purpose of the project is to define and enhance the accuracy and performance of the system. An overview of proposed model development about IoT network and intrusion threat detection using selected features to train and test model are given below. To test and train the dataset of IoT & Network intrusion detection effectively, initially choosing the right dataset for the model. In order to resist the unwanted data displaying, which is irrelevant, inconsistent or incomplete are removed by several steps at preparation. Secondly, feature selection techniques are important to train and test the dataset where the optimal features are identified. Following each stage, four classifiers are selected to evaluate the dataset to determine the malicious traffic flow in the data. Finally, validate the model to show the best performance of each model classifier.

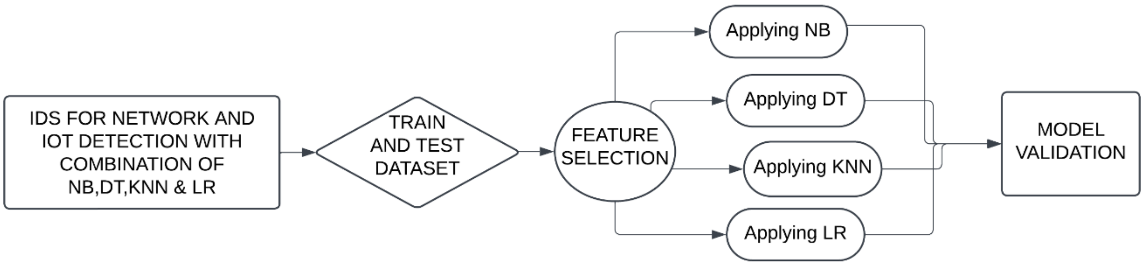


Fig 1: Research Methodology Flow Diagram

3.1 Data Collection

Many different datasets available in public, However, many of these datasets contains inconsistent performance for evaluation. At first, Network intrusion dataset (https://www.kaggle.com/datasets/sampadab17/network-intrusion-detection) consists of isolation in a defence network environment. This dataset includes information on internet control message protocol (ICMP), user datagram protocol (UDP) and transmission control protocols (TCP), used for analyze abstract behavioral patterns. In second dataset, IoT device network logs (https://www.kaggle.com/datasets/speedwall10/iot-device-network-logs/data) contain 6 group, such as Normal, Misconfiguration, DDoS, Site scanning, Host discovery attack and man-in-the-browser attacks.

3.2 Data Pre-processing

This process focuses on balancing the data and avoiding any mistakes on classification which reflects in output of the experiment. Initially, Data are pre-processed to show what are essential and unwanted (Null) values then it prepares for training. Understanding the rows, columns in dataset and removing or replacing null values with nearest fit to obtain efficiency of model. Presented null or invalid values will mislead the model accuracy or performance. In this stage, conversion of string value into numerical attributes to enhance the model. Therefore, the output value of dataset converted into binary values 0 and 1.

3.3 Data Training

The dataset is separated into two sections as train data and test data for evidence based of data mining model. In this model section, the data were split into 70% and 30% for training and testing purposes. Data partitions were used to provide the valuable output value. This will maintain low variations in data and best efficiency with the experiments.

3.4 Model Selection

This section consists of analysis metrics for the research model accuracy, precision, recall and F1 score.

Accuracy: Accuracy is the proportion of correct predictions about the total number of predictions taken on the test set. In this test, the accuracy metric is the most basic. If the dataset be spatially imbalanced, It seems more likely that the model will give a better accuracy if all of the data points are designated as the main class., calculated using:

$$\text{Accuracy} = (TP + TN) / (TP + TN + FP + FN)$$

Precision: The formula for precision is the ratio of true positives to the sum of true positives.

$$\text{Precision} = (TP) / (TP + FP)$$

Recall: The ratio of correctly predicted positive observations to the total actual positives:

$$\text{Recall} = (TP) / (TP) + (FN)$$

F1 Score: Both recall and precision are represented by it, providing the weighted average. While the shortest possible is 0, 1 is the maximum good score and 0 is the minimum one.

$$\text{F1 Score} = 2\times (\text{Precision} \times \text{Recall}) / (\text{Precision} + \text{Recall})$$

4 Design Specification

The section contains a comprehensive description of the design and specification. The source of data implement for these project collected from Kaggle.com, which provides a public dataset. This dataset comprises two sets, namely, "Network Intrusion Data Analysis over TCP/IP" dump data, and it consists of 41 quantitative and qualitative features taken from normal and attack data (3 qualitative and 38 quantitative features) (Bhosale, 2018). The IoT logs dataset has the Ultraviolet detector along GPIO Pins and ESP8266 Wi-Fi module which were carried out to observe the network also obtain the logs (Dixit, 2019). The research uses several supervised ML algorithms to addresses best performance system.

The ML algorithms, namely Naive Bayes, Decision Tree, K-NN, and Logistic regression were employed on the set of both data, and then the results of all the individual tests were combined to identify the detection rate. Then last be situated of detected attacks was then examined, employing the complete input data, the goal is to assess how well integrated IDS model predicts the network and IoT devices attacks.

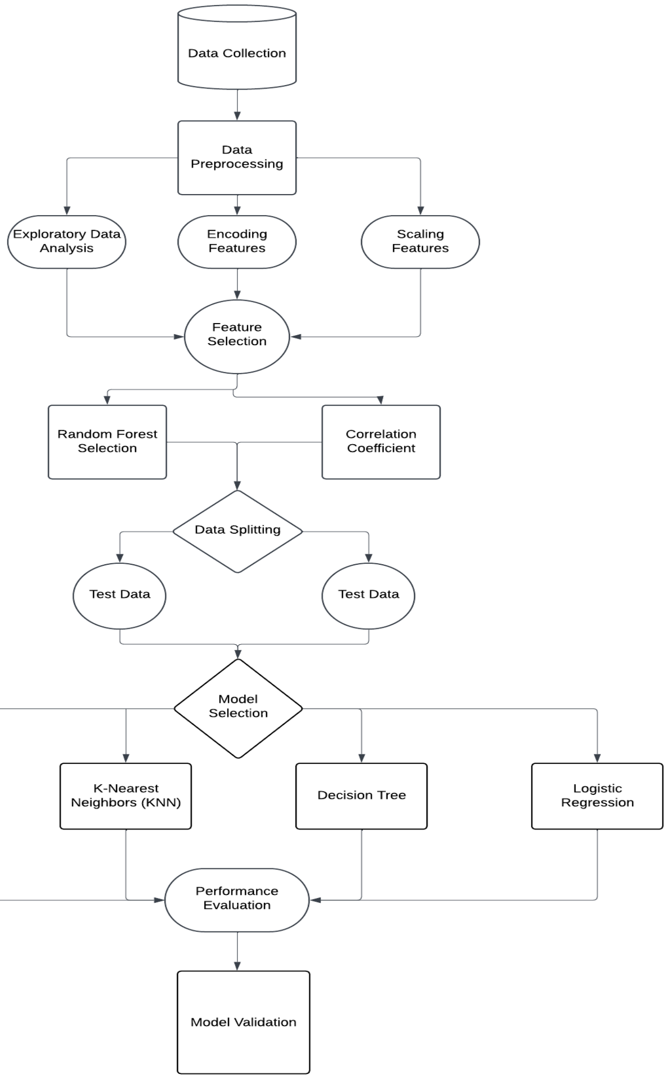


Fig 2: Model design of the IDS for multiple ML models.

4.1 Methods focused on proposed IDS solution

Data Pre-processing

This process is most crucial steps to verify the dataset for evaluating classifiers to assess accuracy and make clear for understanding the predications. Following steps were taken in this stage:

- Cleaning and duplicates of values in the dataset are executed
- Handling missing value in each column and normalize with features
- String values are converted into numerical attributes
- Split the source into 70% and 30% ratio for train and test.
- Splitting the dataset into features label by excluding the attack types and Train-test the split the data.

Data Classification

Four classifiers are performed in isolation to achieve results for individual. The models predicted with parameters like True Positive (TP), False Positive (FP), True Negative (TN) and False Negative (FN). Based on the findings above, the models were described by confusion matrix by calculation of,

- FP: As actual threat is predicted, and the predict is generally it's normal attack.
- FN: These threats are malicious in nature, but the model observed them as normal. Then result that showed to be as false.
- TP: It calculates by what method frequently attacks are correctly identified as positives.
- TN: It determined the number of times attacks were mistakenly judged to be non-existent.

Feature Selection

In this stage, Feature selection is implemented because of determining the most appropriate features in dataset to make it easy to interpret, get accurate results, reduce overfitting, and optimize your computational resources (hex.tech, 2024). In the first dataset, Random Forest feature used due to it's inherent ability to measure the reducing the impurity and improving the spilt quality during the construction of tree. It works on handling non-linear values, works on high dimensional data and resistant to overfitting (Rudnicki, Kursa and Rudnicki, 2011). In the second dataset, correlation coefficient feature selection was applied to find out value for all features in difference with the targeted column and rows. It consisted of 14 features to predict the model, removing high negative value in feature selection from finial result will provide better result.

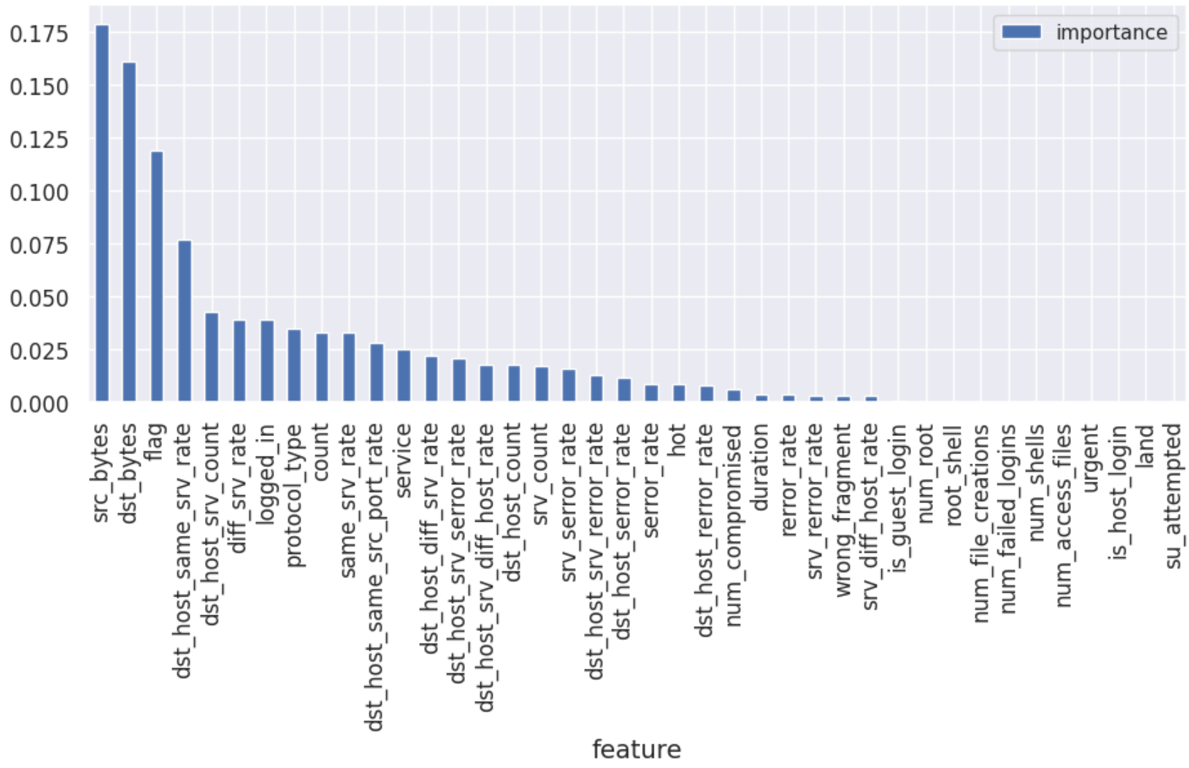


Fig 3: Feature selection random forest classifier on the training set

4.2 Description of the algorithm

Gaussian Naive Bayes

Gaussian Navie Bayes is the kind of Navie Bayes, these methods are supervised learning algorithms applying bayes theorem. It based on random based method it manage every class follow gaussian distribution and predict every parameter has an independent capacity of assuming the value of output. This classification has two variables namely, normal distribution and standard distribution. The standard value of distribution is meant for width of the distribution around the median. Although features are independent of each other's, this might not be applicable for real time detection. In the training process, it focuses on two sides, prior probability and class conditional probability. (Carla Martins, 2023)

The main advantage and disadvantage of Navie Bayes:

- Simple to implement and evaluating the conditional probability is easier.
- Extremely fast- So this technique can be used for where speed training is required.
- In most part, the features show form of dependency. It does not hold conditional independence.
- Zero probability problem. (GeeksforGeeks, 2023)

Decision Tree

Decision Tree are one of the types of supervised learning used for classification and regression. This algorithm will be creating a structure that determine the potential value of attack parameter by understanding of it's decision rules extract from the data factors. A tree can be considered as an approximation of constant. It is a tree-shaped model, as every leaf node represents the results, branches shows the selection of rules and internal leaf node present the attributes. It provides a graph visual representation that shows all specific conditions to issue/choices under certain circumstance. The purpose of using decision tree because of human thinking ability and easier to understand.(Javatpoint.com, 2024)

The various advantages of Decision Tree (DT) are provided below:

- Feasible to interpret and understand. Trees can be visualized.
- Minimal data preparation is needed and other techniques recommended for data normalization, creation of false variable and null values to be removed.
- Able to handle multiple output problem.
- Validate the statistical test and make it possible for reliability of the solution. (scikit-learn, 2024)

K-Nearest Neighbors (KNN)

KNN is the simple supervised ML technique. It predicts the output values form on a group of input values. It categorized the latest set of data based on similar after storing this available data. Because, during recent features is discovered thereafter it is simple to determine into appropriate group. It is utilized for both regression and classification but mostly classified for problems. KNN is also called lazy learner algorithm, which will depend or learn on training data but obtain information from input data as stores the dataset and at time of classification, it preform the results. (Javatpoint, 2024)

The main advantages of using K-NN algorithm are:

- It is faster as takes low training time, because it uses training data as stored and predict the result of data.
- It is easy to perform and execute.

Logistic regression

This model also called as binary logistic regression. It work based on sigmoid function, where the output value is probability and input can be – to + infinity. This helps to clarify the probability event success and failure. It is utilized when the dependent variable is binary (0/1, True/False, Yes/No) in isolation. By evaluating the connection between given set of labled dataset, it supports the categorizing data into discrete classes. The model of linearity within the dependent and independent variables was the primary drawback of this classifier.

The benefits of using logistic regression are:

- It is quick to apply and effective for training purpose for model.
- It can be able to classify fats unknown records.
- low driven to over fitting but at high dimensional dataset it causes overfitting.

5 Implementation

The report section consists of coding samples and creation of guides along with the development for this model performed are showed below:

5.1 Technologies and Software used:

Google colab is the python code editor or also called as IDE used for research. The project setup manual report provides the full detailed discussion about the software and tools are utilized for this research.

5.2 Imported Libraries

- Pandas
- Matplotlib
- Numpy
- Seaborn
- Sklearn
- Imblearn
- Itertools
- confusion matrix
- accuracy score

5.3 Data Preprocessing

This method involves of following steps:

- Implementations like Standard Scaler and Label Encoder were used to extract the numerical attributes and extract categorical values for this dataset.
- Executing Random Forest classifier for its performance on handling high dimensional data effectively, reduce over fitting and avoid from causing imbalanced in dataset.

- Itertools were used at feature selection for generating efficient combination or permutation of feature, it enables the evaluating of various features subsets to identify best model.

5.4 Model Algorithms

To perform and determine the parameters like accuracy, precision, and performance rate, Model algorithms are implemented. The selected classifiers are given below,

- Navie Bayes
- Decision Tree
- k-nearest neighbors (KNN)
- Logistic regression

The predication results of all algorithms are compared with the existing research model. Best two models are classified from the input and provided us the final prediction of research

6 Evaluation

In this section is to provide a effective analysis of the results and main findings of model efficiency is showed. To assess each method and to compare them our proposed model for identifying IoT network intrusion. Model examine with specific algorithm on the Network intrusion and IoT. At the end, high performance classifier is determined by result of experiment.

6.1 Detection of Network and IoT Intrusion Using Naive Bayes Method

The first algorithm is Naive Bayes, which is used for both datasets and their results are shown below.

Dataset	Accuracy	Precision	Recall	F1 score
Network Intrusion dataset	0.90	0.90	0.90	0.90
IoT device network logs	0.57	0.57	0.56	0.48

Table 2: Results of Naïve Bayes

This tabular column shows, Naïve baye classifier performance on network intrusion dataset with obtaining low accuracy and efficiency compared with other algorithm. It indicating classifier found difficult on variability or complexity in IoT network dataset which result reduced predictive effectiveness. As considering large dataset, it does not perform well and consumes more timing to train dataset.

6.2 Detection of Network and IoT Intrusion using Decision Tree Method

The second algorithm is Decision Tree Classifier used for both datasets, their result given below,

Dataset	Accuracy	Precision	Recall	F1 score
Network Intrusion dataset	0.99	0.99	0.99	0.99
IoT device network logs	0.99	0.99	0.99	0.99

Table 3: Results of Decision Tree

It is observed that classifier that performance rate are similar in second dataset. This data has pre-processed prior to use. This result gives near perfect after evaluation and validation the model. However, we found difficult on data imbalanced, overfitting and incorrect target labels. After correcting issue, the predication was accurate and performed well.

6.3 Detection of Network and IoT Intrusion Using K-Nearest Neighbors (KNN) Method

The third algorithm is K-Nearest Neighbors used evaluating both datasets, their result are given below,

Dataset	Accuracy	Precision	Recall	F1 score
Network Intrusion dataset	0.9916	0.9916	0.9916	0.9916
IoT device network logs	0.9994	0.9994	0.9994	0.9994

Table 4: Results of K-Nearest Neighbors

This table 3 shows similar as decision tree but not as same. On network intrusion data it achieved near perfect value, and it reflect as accurate in detection intrusion. Even better it showed in IoT device network logs with achieving 0.99% in all metrics. It indicates reliability and precision in identifying patterns in IoT network logs. As compared to decision tree, K-Nearest Neighbors require more run time than decision tree.

6.4 Detection of Network and IoT Intrusion Using Logistic Regression Method

This fourth algorithm is Logistic Regression used evaluating both datasets, their result are given below,

Dataset	Accuracy	Precision	Recall	F1 score
Network Intrusion dataset	0.95	0.95	0.95	0.95
IoT device network logs	0.88	0.88	0.87	0.87

Table 4: Results of Logistic Regression

This last classifier used for evaluating proposed model. It indicates the model perform well and it reached 0.95% for network detection demonstrating reliable detection capabilities for identifying intrusion accurate. On IoT networks performance slightly lower faced challenges more diversity in IoT data. It shows potential difficulties in generalizing in IoT environments.

6.5 Discussion

In this study, using two datasets, testing was conducted for both binary and multi-label classification. In first dataset, the flaw happens when we mistakenly label a packet as harmful where is literally a legitimate packet. The anomalous package being categorized in 1 and regular package as 0. When packets are wrongly interpreted, when they are anomaly package as usual. This is known as a type II error. In the best case, type II mistakes should be minimized. The second dataset showed five distinct attack types. Normal, Misconfiguration, DDoS, Site scanning, Host discovery attack and man-in-the-browser attacks. As can observed in Table 2, Decision Tree method as received accuracy, precision, recall, and F1 score of 0.99% in first dataset and accuracy, precision, recall, and F1 score of 0.99% on second dataset. As compared with both datasets, the second dataset is significantly better than first dataset. Decision Tree is preferred over the other approaches when it comes to real-time detection.

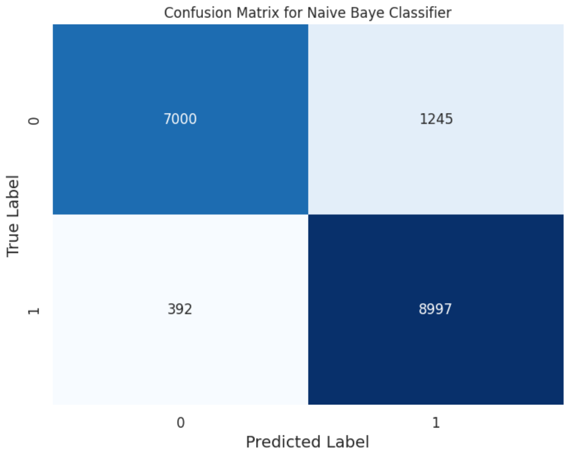


Fig 3: Confusion Matrix for Navie Bayes approach using NID dataset

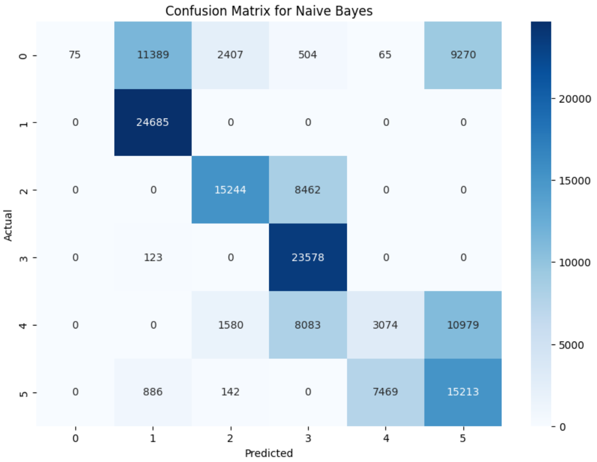


Fig 4: Confusion Matrix for Navie Bayes approach using IoT dataset

In figure 3, shows the network intrusion detection of confusion matrix classifier for Navie bayes in the binary classification prediction of TN has 7000, FP has 1245, FN has 392 and TP has 8997. In the second half of image figure 4, shows the confusion matrix for multi -class classifier task. It denotes that predication of six classes, with significant performance class like 2 and 3, but notable misclassifications like various samples 1 being shows as class 0.

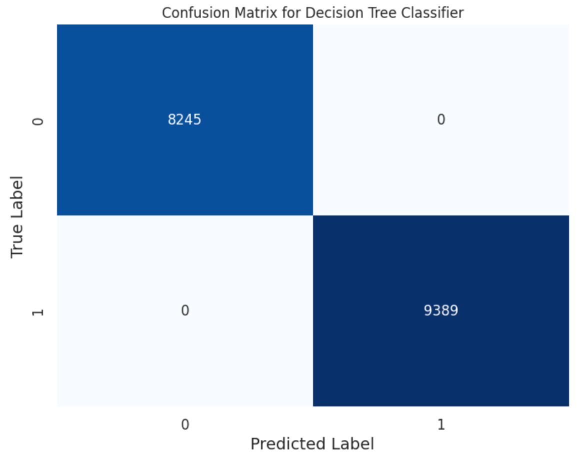


Fig 5: Confusion Matrix for Decision Tree approach using NID dataset

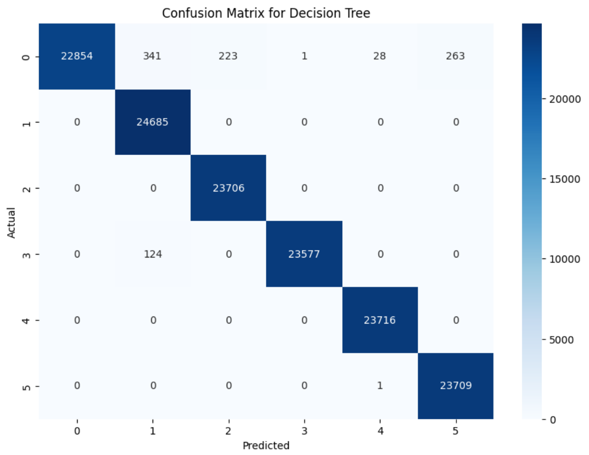


Fig 6: Confusion Matrix for Decision Tree approach using IoT dataset

In figure 5, shows the network intrusion detection of confusion matrix for decision tree were applied to binary classification problem. The predication showed that TN has 8245, FP has 0, FN has 0 and TP has 9389. It indicates that class shows best overall performance with no misclassification for either class (both 0 and 1). This model achieved 99% accuracy of this dataset. In figure 6, shows multi class matrix. The rows represent the actual class, and columns represent the predicated class. Those diagonal values specify the correct predicated values and off diagonal values shows the disarrangements.

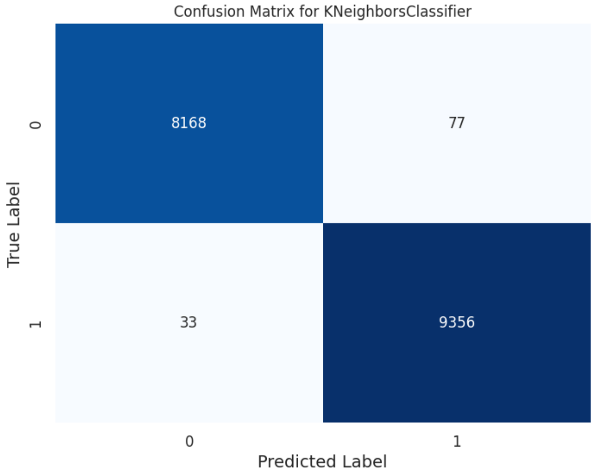


Fig 7: Confusion Matrix for KNN approach using NID dataset

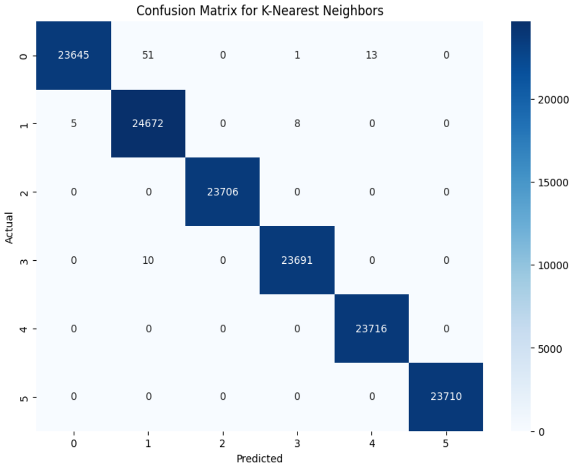
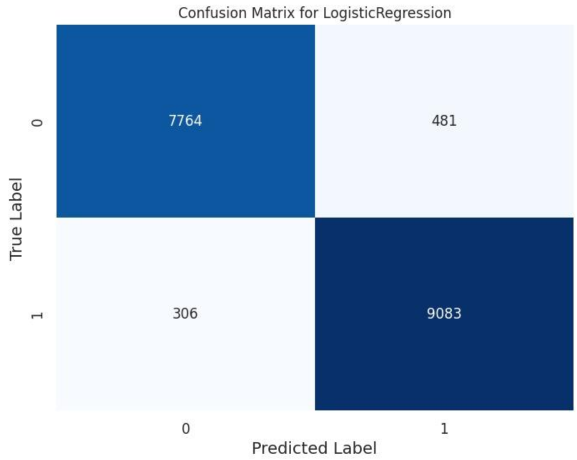


Fig 8: Confusion Matrix for KNN approach using IoT dataset

In figure 7, represents of binary classification matrix with two classes (0 &1) and the correct values (8,168 for class 0 and 9,356 for class 1) control the diagonal. Some of few error values are showed in off-diagonal column. Secondly figure 8, shows the multi class matrix of high dimensional recommends the best performance. Errors such like class 0 & 1 predicting as minimal.



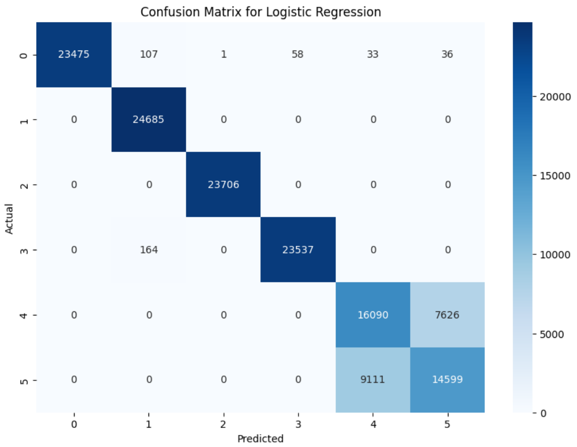


Fig 9: Confusion Matrix for Logistic Regression approach using NID dataset

Fig 10: Confusion Matrix for Logistic Regression approach using IoT dataset

In figure 9, shows the Logistic Regression shows the network intrusion detection of confusion matrix classifier in the binary classification prediction of TN has 7764, FP has 481, FN has 306 and TP has 9083. As model as much better than Navie bayes for network intrusion detection. In figure 10, represents the confusion matrix for multi -class classifier task with same high dimensional performance.

7 Conclusion and Future Work

Applying various machine learning technique, this research Practicum part 2 was conducted with network intrusion dataset and IoT device network log dataset to enhance IoT security. The two datasets resulted with better performance and efficiency concerned. Decision Tree approach had higher accuracy (99.47%) and based on this, helped in answering the research questions "Which is the best machine learning technique can be used to detect both known and unknown network and IoT device logs?" The proposed IDS model is based on Decision Tree and evaluated on two dataset of Network intrusion dataset and IoT device network logs dataset. This model is compared with other three algorithms of Navie Bayes, KNN and the Logistic Regression. The result demonstrates that decision tree has secured the highest accuracy level of 99.47% in the first dataset and 99.31% in second dataset. Additionally, cross validation ensures the model avoids overfitting, class imbalanced and provides reliable performance by testing into multiple data splits. As noticed, KNN also provides nearby better performance of 99.93% in first dataset and 99.16 % second dataset with detection of network intrusion in IoT. The highest accuracy obtained in research was 99.42% & 83% and this model with different algorithms scored accuracy of 99.47% & 99.31% respectively. Hence multiple studies in data analysis, approach and methods of individual models are required to achieve performance. Furthermore, it's been faced heavy to analyse real time data of intrusion within stipulated timeframe and future research work could be considered by using real world intrusion dataset to work on variety of model analysis to find out best performance.

8 References

Amira, A.S., Hanafi, S.E.O. and Hassanien, A.E. (2017) 'Comparison of classification techniques applied for network intrusion detection and classification', *Journal of Applied Logic*, 24, pp. 109–118. Available at: https://doi.org/10.1016/j.jal.2016.11.018.

Asharf, J., Moustafa, N., Khurshid, H., Debie, E., Haider, W. and Wahab, A. (2020) 'A review of intrusion detection systems using machine and deep learning in internet of things: Challenges, solutions and future directions', *Electronics (Switzerland)*. MDPI AG. Available at: https://doi.org/10.3390/electronics9071177.

Bhoi, A.K., Ijaz, M.F., Pramanik, M., Jhaveri, R.H., Chowdhary, C.L., Borah, S. and Panigrahi, R. (2021) 'Performance assessment of supervised classifiers for designing intrusion detection systems: A comprehensive review and recommendations for future research', *Mathematics*. MDPI AG. Available at: https://doi.org/10.3390/math9060690.

Carla Martins (2023) *Gaussian Naive Bayes Explained With Scikit-Learn, https://builtin.com/artificial-intelligence/gaussian-naive-bayes#:~:text=Gaussian%20Naive%20Bayes%20is%20a%20machine%20learning%20classification%20technique%20based,of%20predicting%20the%20output%20variable*.

Choudhary, S., Kesswani, N. and Majhi, S. (2021) 'An Ensemble Intrusion Detection Model For Internet of Things Network'. Available at: https://doi.org/10.21203/rs.3.rs-479157/v1.
Dhanabal, L. and Shantharajah, S.P. (2015) 'A Study on NSL-KDD Dataset for Intrusion Detection System Based on Classification Algorithms', *International Journal of Advanced Research in Computer and Communication Engineering*, 4. Available at: https://doi.org/10.17148/IJARCCE.2015.4696.

Elbasiony, R.M., Sallam, E.A., Eltobely, T.E. and Fahmy, M.M. (2013) 'A hybrid network intrusion detection framework based on random forests and weighted k-means', *Ain Shams Engineering Journal*, 4(4), pp. 753–762. Available at: https://doi.org/10.1016/j.asej.2013.01.003.

Fenanir, S. and Semchedine, F. (2023) 'Smart Intrusion Detection in IoT Edge Computing Using Federated Learning', *Revue d'Intelligence Artificielle*, 37(5), pp. 1133–1145. Available at: https://doi.org/10.18280/ria.370505.

Fu, Y., Du, Y., Cao, Z., Li, Q. and Xiang, W. (2022) 'A Deep Learning Model for Network Intrusion Detection with Imbalanced Data', *Electronics (Switzerland)*, 11(6). Available at: https://doi.org/10.3390/electronics11060898.

GeeksforGeeks (2023) *Gaussian Naive Bayes, https://www.geeksforgeeks.org/gaussian-naive-bayes/*.

Grimaldi, S., Mahmood, A. and Gidlund, M. (2019) 'Real-Time Interference Identification via Supervised Learning: Embedding Coexistence Awareness in IoT Devices', *IEEE Access*, 7, pp. 835–850. Available at: https://doi.org/10.1109/ACCESS.2018.2885893.

Javatpoint (2024) *K-Nearest Neighbor(KNN) Algorithm for Machine Learning, https://www.javatpoint.com/k-nearest-neighbor-algorithm-for-machine-learning*.

Javatpoint.com (2024) *Decision Tree Classification Algorithm*, *https://www.javatpoint.com/machine-learning-decision-tree-classification-algorithm*.

Khraisat, A. and Alazab, A. (2021) 'A critical review of intrusion detection systems in the internet of things: techniques, deployment strategy, validation strategy, attacks, public datasets and challenges', *Cybersecurity*, 4(1). Available at: https://doi.org/10.1186/s42400-021-00077-7.

Krishnan, S., Neyaz, A. and Liu, Q. (2021) *IoT Network Attack Detection using Supervised Machine Learning*.

Liu, Q., Wang, D., Jia, Y., Luo, S. and Wang, C. (2022) 'A multi-task based deep learning approach for intrusion detection', *Knowledge-Based Systems*, 238. Available at: https://doi.org/10.1016/j.knosys.2021.107852.

Liu, Z., Thapa, N., Shaver, A., Roy, K., Yuan, X. and Khorsandroo, S. (2020) 'Anomaly Detection on loT Network Intrusion Using Machine Learning'.
Moustafa, N., Koroniotis, N., Keshk, M., Zomaya, A.Y. and Tari, Z. (2023) 'Explainable Intrusion Detection for Cyber Defences in the Internet of Things: Opportunities and Solutions', *IEEE Communications Surveys and Tutorials*, 25(3), pp. 1775–1807. Available at: https://doi.org/10.1109/COMST.2023.3280465.

Panigrahi, R., Borah, S., Bhoi, A.K., Ijaz, M.F., Pramanik, M., Kumar, Y. and Jhaveri, R.H. (2021) 'A consolidated decision tree-based intrusion detection system for binary and multiclass imbalanced datasets', *Mathematics*, 9(7). Available at: https://doi.org/10.3390/math9070751.

Rai, K., Syamala Devi Professor, M. and Guleria, A. (2015) 'Decision Tree Based Algorithm for Intrusion Detection'.

Rodríguez, E., Valls, P., Otero, B., Costa, J.J., Verdú, J., Pajuelo, M.A. and Canal, R. (2022) 'Transfer-Learning-Based Intrusion Detection Framework in IoT Networks', *Sensors*, 22(15). Available at: https://doi.org/10.3390/s22155621.

Ruzafa-Alcázar, P., Fernández-Saura, P., Mármol-Campos, E., González-Vidal, A., Hernández-Ramos, J.L., Bernal-Bernabe, J. and Skarmeta, A.F. (2023) 'Intrusion Detection Based on Privacy-Preserving Federated Learning for the Industrial IoT', *IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS*, 19(2), p. 1145. Available at: https://doi.org/10.13039/5011000011033.

Sasi, T., Lashkari, A.H., Lu, R., Xiong, P. and Iqbal, S. (2023) 'A comprehensive survey on IoT attacks: Taxonomy, detection mechanisms and challenges', *Journal of Information and Intelligence* [Preprint]. Available at: https://doi.org/10.1016/j.jiixd.2023.12.001.
Satyajit Sinha (2023) *Number-connected-iot-devices, IoT-Analytics (https://iot-analytics.com/number-connected-iot-devices)*.

scikit-learn (2024) *Decision Trees*, *https://scikit-learn.org/1.5/modules/tree.html*.

Thaseen, S. and Kumar, Ch.A. (2013) 'An Analysis of Supervised Tree Based Classifiers for Intrusion Detection System'.