

IJEAST

INTERNATIONAL JOURNAL
OF ENGINEERING APPLIED SCIENCE
AND TECHNOLOGY

VOLUME : 11 ISSUE : 06 Print / Issue Publication Date: October 2026

ISSN : 2455-2143

Indexed In

Google SCHOLAR OPEN ACCESS doi Crossref

WWW.IJEAST.COM

editor@ijeast.com

International Journal of Engineering Applied Sciences and Technology, 2026
Vol. 11, Issue 06, ISSN No. 2455-2143, Pages 21-29
Published Online October 2026 in IJEAST (http://www.ijeast.com)

DETECTION OF THREATS/CYBER ATTACKS IN IOT USING MACHINE LEARNING

Shivangi Dhakad
SCOPE, Vellore Institute of Technology,
Bhopal, Madhya Pradesh, India

Abstract: **In this study, the machine learning models are compared in detecting cyber threats in IoT networks using the Bot-IoT dataset and flow-based features collected by CIC Flow Meter. In the preprocessing stage, various machine learning models were trained and tested to detect various types of attacks after the application of the CST-GR method for feature selection. The results indicate that the number of features in the dataset can be limited to seven of its most important features, while the accuracy remains high and the detection speed is increased significantly, with the Random Forest model performing the best among the tested models. The results in these findings show that optimized feature selection and machine learning are an effective and efficient way for IoT attack detection.**

Keywords: **IoT security, cyber-attack detection, machine learning, intrusion detection, Bot-IoT dataset, feature selection, CST-GR, Random Forest.**

I. INTRODUCTION

The Internet of Things (IoT) is a technology that has ushered in an extremely integrated digital world with millions of devices communicating with very little human interaction. This connectivity is a huge plus for smart cities, healthcare and industry, but it also increases the attack surface for cyber criminals. IoT devices generally feature low processing power, insecure configurations, and different device architectures, which makes them an easy target for attackers. The Mirai botnet is just one example of the risks that can arise from the use of IoT devices in attacks that target multiple nodes. However, the Mirai botnet is just an example of how easily IoT devices can be compromised and how they can be used in the attacks of multiple nodes. With the advancement of IoT, securing IoT systems is now a crucial priority worldwide.

There is a plethora of works in the literature that have studied different machine learning (ML) and deep learning (DL) techniques to enhance IoT security. Previous research has suggested anomaly detection systems, tested various ML models, created new datasets, and shown the benefits of AI-based systems over traditional IDSs, particularly for attacks that change over time. But there are still many obstacles: IoT networks produce heterogeneous traffic; many IDS systems are not capable of detecting attacks specific to the IoT; and many datasets are not diverse enough to create accurate models. A need still exists for optimal feature selection, efficient detection approaches, and models capable of effectively determining a wide range of attack types with speed and scalability in real-time settings for IoT applications.

To address these challenges, the research in this thesis explores the use of machine learning algorithms for cyber-attack detection in IoT networks with the Bot-IoT dataset. The goal is to design an efficient and accurate approach by optimally pre-processing and selecting features. The study is designed to use CICFlowMeter to extract a set of features based on flow and apply the CST-GR feature selection algorithm to select the most relevant features for attack detection and compare the performance of the different ML models when applied to different sets of features. Finally, this work aims to offer a lightweight and reliable framework for detection, which can be applied to real-world IoT systems.

II. RELATED WORK

The Internet of Things (IoT) includes hundreds of millions of connected devices that can communicate with one another with little to no user input. Traditional high-end security solutions are insufficient to secure an IoT system because IoT devices have a limited amount of storage and processing capability. Alsamiri et al.[3] suggested the need for the creation of intelligent network-based security solutions, such as machine learning ones. In this study, we assess several machine learning techniques that can be applied to quickly and accurately identify IoT network assaults. Today, the IoT faces more security challenges than ever before, as it is easily accessible and the demand for smart devices and networks is growing rapidly. There are existing security measures that can be applied to protect the IoT. After discussing the IoT architecture, Tahsien et al.[17] provided an extensive literature review of the ML approach to the importance of IoT security about the potential for different types of attacks. In addition, ML-based solution options for IoT security were presented and future challenges were discussed.

Smart cities work in real-time to improve the comfort and standard of living in metropolitan areas. Since these loT devices are connected to sensors that are directly connected to enormous cloud servers, the network traffic of a smart city via loT systems is expanding tremendously and posing new cybersecurity issues. Alrashdi et al.[2] proposed the Anomaly Detection LoT (AD-IoT) system, a machine learning algorithm-based intelligent anomaly detection system. The

suggested fix can quickly find compromised loT devices at dispersed fog nodes. 2 Our research demonstrated that the AD-loT was capable of achieving the maximum classification accuracy of 99.34 percent with the lowest false positive rate. Following a detailed analysis of the literature on machine learning techniques, Haji et al.[10] analyzed various ML algorithms for attack detection and anomaly detection. It also discussed the significance of IoT security in the context of various potential assaults did. It's a technology that enables billions of intelligent objects, often known as "things," to employ various sensors to gather various kinds of information about themselves and their environment. Once shared, you can use it for a range of things, such as managing and keeping an eye on industrial services, enhanced company services and capabilities, and more. Significant technological advancements in machine learning (ML) have created several new research tools that can be used to address present and upcoming IoT concerns. Ullah et al.[18] gifted a method we've got used to generate a brand new Botnet dataset, from a current one, for anomalous interest detection in IoT networks. The new IoT botnet dataset has a much broader community and flow-primarily based total features. Finally, we used unique gadget studying techniques to check the accuracy of our proposed dataset. We additionally checked the accuracy of our proposed dataset through numerous characteristic correlations and the technique for recursive characteristic elimination. Our proposed IoT botnet dataset offers a floor to research and compares anomalous interest detection versions for IoT networks. We have shared the newly generated Botnet dataset publicly, and a hyperlink is supplied in this paper.

The use of DL for assault detection withinside our online world may be a resilient mechanism to small mutations or novel assaults due to its high-degree function extraction capability. The self-taught and compression abilities of deep studying architectures are key mechanisms for hidden sample discovery from the education information so that assaults are discriminated from benign traffic. Diro et al.[6] aimed toward adopting a brand new approach, deep studying, to cybersecurity to allow the detection of assaults in the social network of things. The overall performance of the deep version is in comparison in opposition to the conventional gadget studying approach, and dispensed assault detection is evaluated in opposition to the centralized detection device. The experiments have proven that our dispensed assault detection device is advanced to centralized detection structures with the usage of a deep studying version. It has additionally been proven that the deep version is extra powerful in assault detection than its shallow counterparts.

Dushimimana et al.[7] implemented a Bidirectional Recurrent Neural Network to construct a protection answer with excessive sturdiness for IoT community protection. Deep getting to know and Machine getting to know have proven terrific bring about handling multimodal and voluminous heterogeneous information about intrusion detection specifically with the structure of recurrent neural networks. In this situation a Random Forest set of rules turned into carried out over Principal Component Analysis due to flexibility, and clean in the usage of the system getting to know the set of rules that permits manufacturing without hyper-parameter tuning, constructing of more than one choice of bushes and merging them collectively to get an extra correct and solid prediction. In this observe a unique set of rules out-finished each RNN and GRNN as it considers each statistics from the beyond and the destiny with again and ahead hidden neurons.

In current years, the usage of the Internet of Things (IoT) has elevated ex ponentially, and cybersecurity issues have elevated in conjunction with it. On the reducing fringe of cybersecurity is Artificial Intelligence (AI), that's used for the improvement of complicated algorithms to shield networks and systems, consisting of IoT systems. However, cyber-attackers have found a way to make the most of AI, and feature even begun to apply antagonistic AI that allows you to perform cybersecurity assaults. Kuzlu et al.[14] compiled statistics from numerous different surveys and studied papers concerning IoT, AI, and assaults with and towards AI and explores the connection among those 3 subjects with the reason of comprehensively imparting and summarizing applicable literature in those fields. The growth of AI/ML has been charged with the development in computing power, mixed with headway in algorithms and cybersecurity isn't an exception. AI is "an area of computer technological know-how that offers giving machines the capacity to appear like they've herbal brilliance." Systems are primarily based totally on AI, every now and then referred to as cognitive structures, are support us automate many roles and equipment up problems which might be greater complicated than maximum human beings are able to solve like New generations of malware and cyber assaults are harder to discover with conventional cybersecurity procedures. Another sizable benefit of AI structures in cybersecurity is that they have unfastened up a massive quantity of time for IT employees. The structures are evolved in this kind of manner that they are capable to act speedy to the state of affairs on their own. And AI structures now no longer make mistakes in finishing their tasks. So in this paper B. Geluvaraj et al.[9] have talked about cybersecurity as a difficulty for this paper and allow us to see the demanding situations and what's the position of AI, ML, and DL in keeping off cybercrimes. The cross-cutting nature of IoT systems and the multidisciplinary components involved in their deployment have created new security challenges. When security mechanisms are implemented to counter a specific security threat, the attackers quickly develop new types of attacks to get around the security measures already in place. To properly safeguard the IoT ecosystem, existing security techniques need be improved. The development of several potent analytical techniques that may be utilised to improve IoT security has been made possible by the advancements in ML and DL. In this study, Mohammed Ali Al-Garadi et al.[1] provided a thorough overview of the potential applications of ML and DL methods in IoT security.

International Journal of Engineering Applied Sciences and Technology, 2026
Vol. 11, Issue 06, ISSN No. 2455-2143, Pages 21-29
Published Online October 2026 in IJEAST (http://www.ijeast.com)

They then discussed how these methods can be used to secure the key IoT layers. At the conclusion, they also provided a lengthy list of issues, difficulties, and potential future directions pertaining to the use of ML and DL in successfully securing IoT systems.

III. METHODOLOGY

A. Background

A.1. Definition

Threat detection is the process of examining a whole security ecosystem to spot any malicious behaviour that might compromise the network. If danger is identified, mitigating measures must be taken to effectively neutralize the threat before it can take advantage of any existing vulnerabilities.

A.2. Working Mechanism

Threat identification notifies you if you are on a secure platform and acts as an early warning system for your computer. By identifying threats, you may stop hackers from accessing your computer and sensitive data from being exposed. Knowing what a cyber danger is necessary before learning how to characterize a threat. A cyber threat, often known as a cybersecurity threat, is any hostile action intended to steal, corrupt, or otherwise interfere with digital life as usual. Computer viruses, data breaches, DoS assaults, and other attack methods are examples of cyber threats. An identification model for cyber threats is built using data from a database of cyberattacks, which is also used to train the model with knowledge of prior assaults. Threats are then detected using the prediction model.

A.3. Workflow Diagram

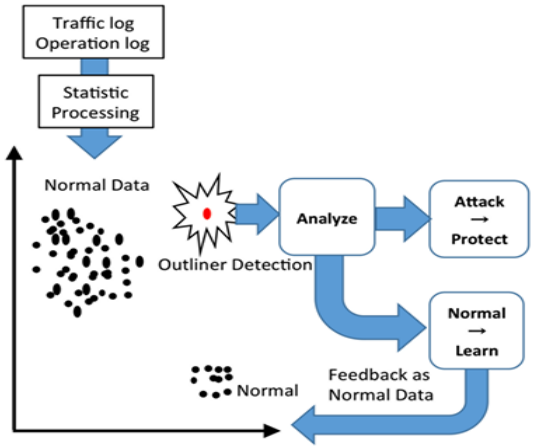


Figure 1: Flow of cyber attack detection system

A.4. Types of cyber threats

• DDoS: Since the attacker targets systems, servers, or networks, this poses a serious danger to enterprises. In order to deplete their resources and bandwidth, they then bombard them with traffic. An alternative name for this assault is a DDoS (Distributed Denial-of-Service) attack.

• CATO: Corporate Account Takeover (CATO) is the practice of online criminals pretending to be a company in order to execute fraudulent wire transfers and ACH transactions and steal company assets. Transfers of unauthorised monies are made to accounts controlled by cybercriminals.

• Malware Attacks: One of the most prevalent kinds of cybersecurity attacks is this one. Viral malware is referred to in this sentence. Examples include worms, spyware, ransomware, adware, and trojans. Malware breaks into a network and opens email attachments in addition to downloading them.

• Password Attack: In this attack, a user's password is decrypted using a number of applications and tools, including Aircrack, Cain, Abel, John the Ripper, Hashcat, etc.

• SQL Injection Attack: On a website that uses a database, this form of assault takes place. By inserting malicious code into a search field on a susceptible website, the hacker manipulates a typical SQL query. Thus, the attacker has access to read, change, and destroy database tables.

A.5. Software tools available to implement cyber threat

• SolarWinds Security Event Manager: An intrusion detection solution for networks and hosts is SolarWinds Security Event Manager. It monitors, responds to, and reports security risks in real-time. It provides robust log search functionality. It is a scalable cloud-based system.

• Intruder: The most well-known cloud-based network vulnerability scanner is called Intruder, and it may help you uncover cybersecurity flaws in your most vulnerable systems to prevent expensive data breaches. Your cybersecurity problems can be resolved with it. Your time is greatly reduced as a result.

• Acunetix: The answer to securing your websites, online apps, and APIs is Acunetix. This application security testing tool can scan all sites, web apps, and sophisticated web applications and detect over 7K vulnerabilities. It includes capabilities for managing vulnerabilities built right in. Acunetix offers both on-premises and on-demand deployment options.

• Invicti (formerly Netsparker): Enterprises may test their application security using Invicti. Throughout the SDLC, it provides the capabilities 6 and functions needed to automate security testing. Automation, visibility, accuracy, scalability, and security are all capabilities of Invicti.

• Webroot: A cloud-based platform is used by Webroot. It can defend mobile devices, Macs, and PCs. It offers a solution for partners, enterprises, home offices, and personal usage. Platforms for Windows, Mac, Android, and iOS are supported.

• Syxsense: By combining security scanning, patch management, and remediation into a single cloud-based interface, Syxsense Secure enables IT and security teams to halt breaches with a single endpoint security solution.

• CIS: The Centre for Internet Security is referred to as CIS. It offers a range of memberships, services, and cybersecurity

International Journal of Engineering Applied Sciences and Technology, 2026
Vol. 11, Issue 06, ISSN No. 2455-2143, Pages 21-29
Published Online October 2026 in IJEAST (http://www.ijeast.com)

technologies. It offers CIS SecureSuite for business use. CIS Benchmarks and CIS Controls are both part of the CIS Security package.

A.6. Role of Machine Learning in Cyber threat detection

Based on past behaviour, ML enables IoT security teams to make smart predictions and reactions. It correlates current network activity with behaviour patterns from attack samples in the case of known vulnerabilities and attacks, including distributed denial of service, and takes preventative action.

ML capabilities for IoT security, including device-level anomaly detection and automated threat response, are provided by services like AWS IoT Device Defender, Extreme Networks products, or Microsoft's Azure Security Centre for IoT.

In the Windows Defender by Microsoft illustration, client-side and cloud-based ML algorithms simultaneously assess current network use against 30 security protection models concurrently. To decide whether a known assault would behave in a good or negative way, some of those models consider millions of variables.

IoT devices and network activity are monitored by ML models to spot unusual behaviour in real time and take appropriate action to guard against undiscovered vulnerabilities and zero-day attacks. Because so many ML solutions continuously update every day to stay up with the evolving threat landscape, ML is perfect for securing intricate networks. It quickly analyses an IoT fleet's extensive digital trail and contrasts its actions with known dangers and past patterns of behaviour. Only a network with ML systems can react thus swiftly to identify threats before they use IoT devices to breach the primary corporate network.

B. Problem Statement

This section analyzes the IOT cyber attack threat model and research issues. IoT devices have exploded with multiple heterogeneous devices. This leads to increased IoT vulnerabilities (e.g., zero-day attacks, botnets, etc.) that serve attack vectors that can target IoT victims and use them to steal personal information or launch DDoS attacks over the Internet. For example, IoT botnets can make it easy to compromise vulnerable IoT devices by scanning the Internet and finding the IP address of the victim's IoT network. The Mirai botnet phenomenon occurred in late 2016 when it compromised a large number of IoT devices by launching a DDoS attack against DNS provider Dyn [4]. That's why these IoT systems can take years to build but only days to fail. Major challenges in securing IoT devices can be IoT botnets without notice to ISPs and eligible users. We believe that the security problems of most smart home networks are increased because they are built with vulnerable and poor security technologies, because some companies need to take advantage or have little security knowledge. The main security issues of massive IoT devices are built on heterogeneous devices and protocols, which are very difficult to protect in a smart city [11]. Therefore, most IoT networks are not satisfied with traditional IDS systems that do not accurately detect IoT attack networks [20]

C. Proposed solution

The dataset utilised and our suggested method for identifying attacks in IoT networks are briefly described in this section. The method we provide uses a variety of pre-processing steps as well as actual applications to identify abnormalities using machine learning methods. The raw dataset was first processed using CI CFlowMeter [15] to extract flow-based characteristics. After pre-processing the data, the dataset was split into two portions for training and testing in the first stage. Pre-processing of the data is necessary to get it into a form that machine learning algorithms can use. Following these activities, the feature selection stage determines the attributes that will be employed by the algorithms. The application of machine learning techniques completes our strategy. Because of its frequent updates, extensive attack variety, inclusion of IoT-generated traffic, and capability to produce new characteristics from the raw dataset, the Bot-IoT dataset was chosen for the trials. The Australian Centre for Cyber Security's Cyber Range Lab produced the Bot-IoT dataset. (ACCS). Based on botnet scenarios, this dataset contains three basic types of assaults: probing, denial-of-service attacks, and information theft. From the unprocessed traffic traces, we extracted flow-based characteristics using CICFlowMeter. To produce 84 network traffic characteristics, CIC distributes CICFlowMeter [19], a network traffic flow generator.

C.1. Architecture

The IoT architecture serves as a portal to various hardware applications to establish connections and extend IoT's services at each entryway. Different networking protocols are used to transmit and receive information/data from different layers of the IoT architecture, including Bluetooth, Wi-Fi, RFID, narrow and broadband, ZigBee, and LPWAN. The physical, network, and application layers make up the three primary levels of an IoT architecture.

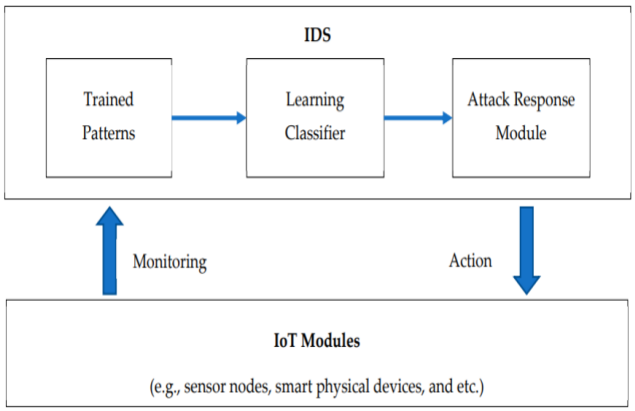


Figure 2: Architecture of the model

International Journal of Engineering Applied Sciences and Technology, 2026
Vol. 11, Issue 06, ISSN No. 2455-2143, Pages 21-29
Published Online October 2026 in IJEAST (http://www.ijeast.com)

• Trained Patterns: It displays the usual IoT environment intrusion detection system. The detection system keeps an eye on the IoT components, and as soon as it determines that the traffic pattern is an attack, it will send an alarm to the users or the reaction modules to process against the suspected threats. For the purpose of developing their detection systems, each of the aforementioned systems has to record the regular traffic patterns or/and the attack signature patterns.
• Learning Classifier: An appropriate learning classifier must be used to train the detection model in the machine learning-based technique. To evaluate whether or if the incoming traffic pattern constitutes an attack, almost all misuse-based techniques employ a pattern matching algorithm.
• Attack Response Module: The detection system keeps track of the IoT components, and as soon as it determines that the traffic pattern is an attack, it will send an alert to the users or the reaction modules to process against the suspected threats.
• IoT Modules: An Internet of Things (IoT) module is a tiny electronic gadget placed in machines, equipment, and things that connects to wireless networks and transmits and receives data. IoT situations can be extremely demanding because of their severe humidity, vibration, and temperature. No matter the location or wireless network, IoT wireless modules must be ruggedised to deliver dependable connectivity without service interruption.

IV. IMPLEMENTATION

System Requirement:
OS- Windows/ubuntu
RAM-16GB
CPU– Intel Core i7, 8 cores
Language- Python

A. Algorithm For The Model

We introduce our new feature selection technique, the CST-GR algorithm, in this part to help us choose the best characteristics for identifying each type of specialised assault. Algorithm 1 displays the suggested feature selection algorithm. To identify the most highly correlated feature collection, the merit function from the CFS method [35] is used. Such characteristics do not necessarily exist in our final feature set, unlike the CFS algorithm. To pick the final feature, the gain-ratio is also computed for each of those characteristics. In order to pick the final features from the initial feature set, the minimal gain-ratio value in the feature set generated using the merit function is employed as the threshold of gain-ratio.

Algorithm 1: Correlated-Set Thresholding on Gain-Ratio (CST-GR)
Input: Feature Set (FI)
Output: Selected Feature Set (FS)

• Read all the original features to the set FI. FI = f1, f2, f3,..., fn, n = the number of the original features.
• Extract the feature set having the highest correlation based on the merit function: Equation (1), FC subset FI FC = f1, f2,..., fc, c = the number of features in the subset having the best merit value.
• Calculate the minimum gain value of the features in FC, f min (FC)
• Use Equation (2) to calculate the gain-ratio value of each feature of FI GR = f1, v1, f2, v2,..., fn, vn, where vi is the gain-ratio value of fi(1 greater than equal i less than equal n)
• From GR, select the features whose gain-ratio values are greater than or equal to f min (FC), FS subset FI FS = r1, r2, r3,....., rs, s = the number of finally selected features
• Output FS

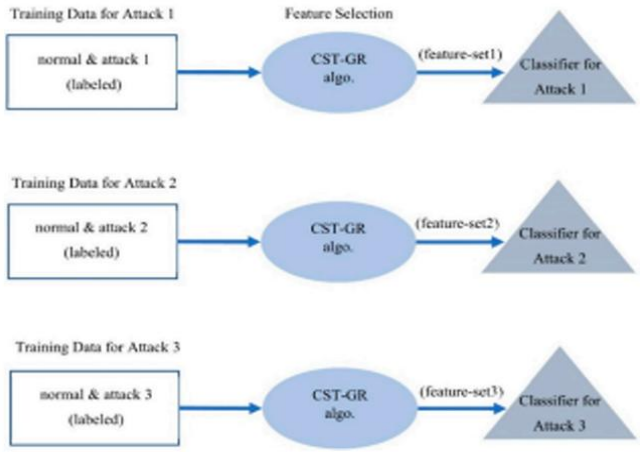


Figure 3: The training phase (the case of three classifiers). CST-GR, correlated set thresholding on gain-ratio.

B. Steps of Implementation

• Extraction of Features: From the raw network traffic data, flow-based features (in pcap format) were extracted using CICFlowMeter[15]. CICFlowMeter is a network traffic flow generator that generates 84 network traffic characteristics and is sold by CIC. It reads the pcap file, extracts the features, and creates a visual document of them. It also provides a CSV file containing the dataset. By extracting fresh dataset characteristics, this approach was primarily intended to increase the prediction power of classifiers.
• Data preprocessing: The dataset is transformed using preprocessing data transformation operations into a structure suitable for machine learning. Cleaning the dataset in this stage also makes it more effective by eliminating faulty or unnecessary data that might impair the dataset's accuracy.
• Data splitting: In order for machine learning to function, data must be available.[13] In addition to the data needed for training, test data are also necessary to assess the algorithm's performance and determine how well it performs. In our study, we divided the Bot-IoT dataset into two categories: training data (80percent) and testing data (20percent).

International Journal of Engineering Applied Sciences and Technology, 2026
Vol. 11, Issue 06, ISSN No. 2455-2143, Pages 21-29
Published Online October 2026 in IJEAST (http://www.ijeast.com)

• Feature Selection: To create a simple security solution suitable for IoT systems[16], it is important to reduce the number of characteristics and only utilise those required to train and test the algorithms. As a method for selecting features, we employed the Random Forest Regressor algorithm. It has been demonstrated that a dataset's dimensions may be reduced effectively using the random forest regressor. The model trains and reacts more quickly when the input data characteristics are reduced from more than 80 network traffic features to 7.
• Implementation of Machine Learning Algorithms: Python machine learning libraries were used throughout all of the studies (scikit-learn, Matplotlib, Pandas, and NumPy). We divided the evaluation of machine learning algorithms for the dataset into three stages: applying the proposed algorithms separately to each attack in the dataset, applying the algorithms to the entire dataset with a set of features combining the best features for each attack (the list of these features can be found in Table II), and applying the algorithms to the entire dataset with the seven best features found during the feature selection stage.

C. Code for the model

```
data = read_all_data()
def number_str(number: int):
    return  "\\numprint" + "{" + "{}".format(number) + "}"
def count_class_samples(device_id: int):
    binary_dataset = get_dataset(data[device_id:device_id+1], multiclass=False)
    binary_labels = binary_dataset[:][1].numpy()
    features = binary_dataset[:][0].numpy()

    unique, counts = np.unique(binary_labels, return_counts=True)
    percentage_benign = counts[0] * 100 / (counts[0] + counts[1])
    percentage_attack = counts[1] * 100 / (counts[0] + counts[1])
    print("Number of benign instances: " + number_str(counts[0]) + " ({:.2f}\\%)".format(percentage_benign))
    print("Number of attack instances: " + number_str(counts[1]) + " ({:.2f}\\%)".format(percentage_attack))
    print("Total number of instances: " + number_str(counts[0] + counts[1]))
    return counts[0], counts[1]
min_benign = 1e9
min_attack = 1e9
sum_benign = 0
sum_attack = 0

for device_id in range(9):
    print('Device {}'.format(device_id + 1) + ': ' + all_devices[device_id])
    benign, attack = count_class_samples(device_id)
    min_benign = min(min_benign, benign)
    min_attack = min(min_attack, attack)
    sum_benign += benign
    sum_attack += attack
    print()

print("Minimum number of benign samples: " + number_str(min_benign))
print("Minimum number of attack samples: " + number_str(min_attack))

percentage_benign = 100 * sum_benign / (sum_benign + sum_attack)
percentage_attack = 100 * sum_attack / (sum_benign + sum_attack)

print("Total number of benign samples: " + number_str(sum_benign) + " ({:.2f}\\%)".format(percentage_benign))
print("Total number of attack samples: " + number_str(sum_attack) + " ({:.2f}\\%)".format(percentage_attack))
print("Total number of samples: " + number_str(sum_benign + sum_attack))
```

Output for the code :

```
Device 1: Danmini_Doorbell
Number of benign instances: \numprint{49548} (4.87\%)
Number of attack instances: \numprint{968750} (95.13\%)
Total number of instances: \numprint{1018298}

Device 2: Ecobee_Thermostat
Number of benign instances: \numprint{13113} (1.57\%)
Number of attack instances: \numprint{822763} (98.43\%)
Total number of instances: \numprint{835876}

Device 3: Ennio_Doorbell
Number of benign instances: \numprint{39100} (11.00\%)
Number of attack instances: \numprint{316400} (89.00\%)
Total number of instances: \numprint{355500}

Device 4: Philips_B120N10_Baby_Monitor
Number of benign instances: \numprint{175240} (15.95\%)
Number of attack instances: \numprint{923437} (84.05\%)
Total number of instances: \numprint{1098677}
```

V. EVALUATION

A. Evaluation Metrics

It is essential to identify performance metrics that are appropriate for the problem to be solved when assessing the performance of machine-learning models. The most crucial performance measures for accuracy, precision, f-measure, and recall were utilised to assess our findings and are represented in the equations below:

$$Precision = \frac{TP}{TP + FP} \quad (1)$$

$$Recall = \frac{TP}{TP + FN} \quad (2)$$

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (3)$$

$$F - measure = \frac{2}{\frac{1}{Recall} + \frac{1}{Precision}} \quad (4)$$

B. Results

As mentioned in the section above, we divided the assessment of machine learning algorithms for the dataset into the following three stages. Phase 1 involves applying machine learning algorithms to each attack in the dataset separately; Phase 2 involves applying machine learning algorithms to the whole set using a set that combines the best features from each attack; and Phase 3 involves applying machine learning algorithms to the whole set using the top seven features determined during the feature selection step.[12] The following tables list the outcomes of all the experiments. The performance evaluation processes were carried out ten times for each machine learning method, and the values in the tables represent the arithmetic means of these ten processes.
Phase 1: The first step is to individually apply machine learning techniques to each assault in the dataset.[5] Ten

International Journal of Engineering Applied Sciences and Technology, 2026
Vol. 11, Issue 06, ISSN No. 2455-2143, Pages 21-29
Published Online October 2026 in IJEAST (http://www.ijeast.com)

different attack types are subjected to seven different machine learning techniques, and the results are shown in the table below. If there is an equality in the F-measure in the algorithmic results, the following values are checked to rule out equality: precision, accuracy, recall, and time.

Attack Names	**F-Measures**						
	NB	**QDA**	**RF**	**ID3**	**AB**	**MLP**	**KNN**
DDOS_HTTP	0.72	0.85	**0.96**	**0.96**	0.96	0.95	0.96
DDOS_UDP	0.73	0.92	**0.98**	**0.98**	0.98	0.97	0.98
DDOS_TCP	0.71	0.85	0.99	0.99	**1.0**	0.74	0.99
DOS_HTTP	0.72	0.82	**0.96**	**0.96**	0.95	0.95	0.96
DOS_UDP	0.72	0.83	0.97	0.97	**0.98**	0.98	0.97
DOS_TCP	0.64	0.74	**1.0**	**1.0**	1.0	0.78	0.99
Data exfiltration	0.72	0.76	**0.97**	**0.97**	0.97	0.94	0.97
Keylogging	0.72	0.82	0.95	0.95	0.95	0.91	**0.98**
Service_Scan	0.73	0.83	**0.95**	**0.95**	0.95	0.94	0.94
OS_Scan	0.72	0.76	0.97	0.97	0.98	0.97	**0.99**

Phase 2: Using a collection of features that combines the best attributes for each assault, machine learning methods are applied to the whole dataset. In this step, the entire dataset is utilised. On the total dataset, seven distinct machine learning techniques were applied, and we made use of feature sets that were individually extracted for each assault. Table IV displays the outcomes from the use of the 13 attributes that were retrieved from the assaults.

ML Algorithm	**Accuracy**	**Precision**	**Recall**	**F-Measure**	**Time**
NB	0.78	0.84	0.78	0.75	**5.056**
QDA	0.88	0.89	0.88	0.87	6.1964
RF	0.98	0.98	0.98	0.98	27.0328
ID3	0.99	0.99	0.99	0.99	19.3447
Adaboost	1.0	1.0	1.0	**1.0**	308.9403
MLP	0.84	0.88	0.84	0.83	1011.5001
KNN	0.99	0.99	0.99	0.99	2052.1801

Phase 3: This involves using machine learning techniques to analyse the full dataset using the top seven characteristics found during the feature selection process. The performance of the algorithms did not significantly alter in terms of F-measure, but in terms of speed, all of the algorithms' running times were dramatically shortened. Because 13 characteristics are employed in the procedure applied in Table V, as opposed to just 7 attributes in Table IV, there is a reduction in execution time. The machine learning algorithms ran faster due to the decrease in feature count.

ML Algorithm	**Accuracy**	**Precision**	**Recall**	**F-Measure**	**Time**
NB	0.79	0.85	0.79	0.77	**4.0472**
QDA	0.87	0.89	0.87	0.86	4.4056
RF	0.97	0.97	0.97	0.97	28.9246
ID3	0.97	0.97	0.97	0.97	17.0899
Adaboost	0.97	0.97	0.97	0.97	238.8618
MLP	0.84	0.87	0.84	0.83	949.6977
KNN	0.99	0.99	0.99	**0.99**	1615.9852

The final outcomes of the implementation are contrasted with research from the literature (see Table VI). The study completed in 2019 by Ferrag et al.[8] was selected for this comparison. This is because the previously stated research utilised the same dataset as well as two machine learning techniques that were identical to the ones we employed. The machine learning methods Random Forest and Naive Bayes are comparable. The feature set utilised distinguishes our work from theirs in a significant way. They utilised the original feature set, but we utilised a fresh feature set that CICFLOWMETER extracted. The primary criterion for evaluation was decided to be the detection rate (Recall). The

comparison of the findings from the two investigations is shown in the table. Examining the data reveals that the Random Forest method utilised in our study performs better than that in[8], while the NB approach performs similarly for the majority of attack types. So, it is clear that the performance of both algorithms was improved by the new characteristics employed in our research.

Attack Names	**Ferrag et al[14]**		**Our Work**	
	RF	**NB**	**RF**	**NB**
DDOS_HTTP	82.26%	50.78%	96%	71%
DDOS_UDP	88.28%	78.67%	99%	70%
DDOS_TCP	55.26%	78.50%	98%	72%
DOS_HTTP	82.20%	68.68%	95%	71%
DOS_UDP	81.77%	65.56%	100%	63%
DOS_TCP	82.99%	100%	97%	71%
Data exfiltration	86.55%	66.55%	96%	71%
Keylogging	70.12%	65.62%	95%	71%
OS_Scan	82.20%	68.68%	94%	70%
Service_Scan	69.82%	65.21%	95%	72%

In the above table, we can see the comparison of the performance of the two algorithms with our work

VI. CONCLUSION AND FUTURE WORK

This study presented a machine learning–based approach for detecting IoT cyber-attacks using the Bot-IoT dataset. By extracting flow-based features with CICFlowMeter and applying the CST-GR feature selection method, the feature set was reduced from over 80 attributes to seven key features, improving efficiency without loss of accuracy. Experimental results showed that the Random Forest classifier achieved the highest overall performance. The findings demonstrate that optimised feature engineering combined with machine learning provides an effective and lightweight solution for IoT intrusion detection.

Several avenues for future research remain. First, deep learning architectures such as LSTM, GRU, or hybrid neural network models may further enhance detection performance, particularly for temporal or complex attack behaviours. Second, deploying the optimised model on edge or fog devices would allow assessment of its real-time performance and practicality under resource constraints. Third, evaluating the framework on additional IoT datasets would help validate its generalizability.

Additionally, incorporating adversarial learning or anomaly-based methods may improve resistance to zero-day and adversarial attacks. Future studies may also explore hybrid feature selection approaches combining CST-GR with evolutionary or embedded methods to refine feature subsets. Ultimately, the development of a comprehensive intrusion detection system, encompassing dashboards, automated alerts, and response modules, would facilitate real-world deployment.

VII. REFERENCES

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

[2] I. Alrashdi, A. Alqazzaz, E. Aloufi, R. Alharthi, M. Zohdy, and H. Ming. (2019). Ad-iot: Anomaly detection of iot cyberattacks in smart city using machine learning. In 2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC), pages 0305–0310. IEEE.

[3] J. Alsamiri and K. Alsubhi. (2019). Internet of things cyber attacks detection using machine learning. International Journal of Advanced Computer Science and Applications, 10(12).

[4] M. Antonakakis, T. April, M. Bailey, M. Bernhard, E. Bursztein, J. Cochran, Z. Durumeric, J. A. Halderman, L. Invernizzi, M. Kallitsis, et al. (2017). Understanding the mirai botnet. In 26th USENIX Security Symposium (USENIX Security 17), pages 1093–1110.

[5] I. Cviti´c, D. Perakovi´c, M. Periˇsa, and M. Botica. (2021). Novel approach for detection of IoT-generated DDoS traffic. Wireless Networks, 27(3):1573–1586.

[6] A. A. Diro and N. Chilamkurti. (2018). Distributed attack detection scheme using deep learning approach for internet of things. Future Generation Computer Systems, 82:761–768.

[7] A. Dushimimana, T. Tao, R. Kindong, and A. Nishyirimbere. (2020). Bi-directional recurrent neural network for intrusion detection system (ids) in the internet of things (iot). International Journal of Advanced Engineering Research and Science, 7(3):524–539.

[8] M. A. Ferrag and L. Maglaras. (2019). Deepcoin: A novel deep learning and blockchain-based energy exchange framework for smart grids. IEEE Transactions on Engineering Management, 67(4):1285–1297.

[9] B. Geluvaraj, P. Satwik, and T. Ashok Kumar. (2019). The future of cybersecurity: Major role of artificial intelligence, machine learning, and deep learning in cyberspace. In International Conference on Computer Networks and Communication Technologies, pages 739–747. Springer.

[10] S. H. Haji and S. Y. Ameen. (2021). Attack and anomaly detection in iot networks using machine learning techniques: A review. Asian journal of research in computer science, 9(2):30–46.

[11] M. M. Hossain, M. Fotouhi, and R. Hasan. (2015). Towards an analysis of security issues, challenges, and open problems in the internet of things. In 2015 ieee world congress on services, pages 21–28. IEEE.

[12] O. Ibitoye, O. Shafiq, and A. Matrawy. (2019). Analyzing adversarial attacks against deep learning for

intrusion detection in iot networks. In 2019 IEEE global communications conference (GLOBECOM), pages 1–6. IEEE.

[13] N. Koroniotis, N. Moustafa, E. Sitnikova, and J. Slay. (2017). Towards developing network forensic mechanism for botnet activities in the iot based on machine learning techniques. In International Conference on Mobile Networks and Management, pages 30–44. Springer.

[14] M. Kuzlu, C. Fair, and O. Guler. (2021). Role of artificial intelligence in the internet of things (iot) cybersecurity. Discover Internet of things, 1(1):1–14.

[15] A. H. Lashkari, G. Draper-Gil, M. S. I. Mamun, A. A. Ghorbani, et al. (2017). Characterization of tor traffic using time based features. In ICISSp, pages 253–262, 2017.

[16] M. K. Putchala. (2017). Deep learning approach for intrusion detection system (ids) in the internet of things (iot) network using gated recurrent neural networks (gru).

[17] S. M. Tahsien, H. Karimipour, and P. Spachos. (2020). Machine learning based solutions for security of internet of things (iot): A survey. Journal of Network and Computer Applications, 161:102630.

[18] I. Ullah and Q. H. Mahmoud. (2020). A technique for generating a botnet dataset for anomalous activity detection in iot networks. In 2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC), pages 134 140. IEEE.

[19] S. Yu. (2018). Study on the internet of things from applications to security issues. In International Conference on Cloud Computing and Security, pages 80 89. Springer.

[20] B. B. Zarpel˜ao, R. S. Miani, C. T. Kawakani, and S. C. de Alvarenga. (2017). A survey of intrusion detection in internet of things. Journal of Network and Computer Applications, 84:25–37.

IJEAST

INTERNATIONAL JOURNAL OF ENGINEERING APPLIED SCIENCE AND TECHNOLOGY

AUTHORS

Shivangi Dhakad

ABOUT IJEAST

International journal of Engineering Applied Science and Technology (IJEAST) is a peer-reviewed, open access journal that publishes high - quality research paper in the field of Engineering, Applied Science and Technology.

IJEAST aims to provide a platform for researchers, academicians, and professionals to share their innovative ideas, research findings, and practical experiences with the global scientific community.

JOURNAL METRICS

H-INDEX	i10-INDEX	IJEAST CITATIONS	PEER-REVIEWED & OPEN ACCESS
33	154	9,647 (IJEAST CITATIONS)	PUBLISHED MONTHLY

PEER REVIEWED

All submission are rigorously peer reviewed to ensure quality.

AUTHORS ARE OUR PRIORITY

We value our authors and recognize their contribution to the advancement of research and knowledge.

OPEN ACCESS

Free and unrestricted access to research for all.

GLOBAL REACH

Connecting researchers and professionals worldwide.

TIMELY PUBLICATION

We ensure a swift and efficient publication process.

For more information, visit our website
www.ijeast.com

editor@ijeast.com
www.ijeast.com
India

2455-2143