Universal Library of Engineering Technology Research Article

ISSN: 3064-996X

Open Access | PP: 42-52

DOI: https://doi.org/10.70315/uloap.ulete.2023.007

# An AI-Based Framework for Detecting IoT Botnets Through Network Traffic Analysis and Modeling

**Srikanth Reddy Keshireddy[1], Venkata Teja Nagumotu[2], Harsha Vardhan Reddy Kavuluri[3], Akhil Kumar Pathani[4], Ajay Dasari[5], Venkata Kishore Chilakapati[6]**

[1]Senior Software Engineer, Keen Info Tek Inc.
[2]Sr Network Engineer, Techno-bytes Inc.
[3]Lead database administrator, Wissen infotech Inc.
[4]Sr Network Engineer, Ebay.
[5]Senior Support Engineer, Microsoft.
[6]Technical Advisor, Microsoft.

## Abstract

*The Internet of Things (IoT) has become a significant cybersecurity issue, with botnets such as Gafgyt and Mirai carrying out large Distributed Denial of Service (DDoS) attacks against smart devices by taking advantage of lax security standards. The conventional machine learning approaches have shown relatively low performance in identifying complex botnet attacks since they are incapable of learning complicated temporal relationships among network traffic patterns. The study overcomes these drawbacks by proposing a hybrid Bi-LSTMGRU deep learning model (DLM) that can detect IoT botnets effectively. The methodology uses the full N-BaIoT dataset of nine IoT device traffic with 11 classes (benign, 10 attack variants) and uses systematic preprocessing, including NaN removal, duplicate removal, and MinMax normalization. The hybrid architecture is a combination of bidirectional LSTM and GRU in a synergetic manner with the temporal dependency learning and sequential processing efficiency, respectively, to glean out more complex traffic patterns. The experimental results are exceptional, having an accuracy of 98.96, precision of 98.67, recall of 99.02 and F1-score of 98.56. The analysis of ROC-AUC confirms a great level of discrimination with seven classes reaching the score of 1.00 (AUC) and the rest of the classes reaching a score of over 0.97. The comparative evaluation demonstrates that substantial superiority is better than the current methods: KNN (86.98%), Random Forest (69.49%), ANN (75%), and the Naive Bayes (93.2) have improved accuracy by 5-29%. The presented work provides a state-of-the-art solution to real-time IoT security monitoring and proves that hybrid DL architectures are effective in securing smart device ecosystems. The framework provides a scalable base of real-world intrusion detection systems, as well as the determination of future directions such as cross-dataset validation, edge device optimization, explainable AI integration, and adaptive learning mechanisms of new threats.*

**Keywords:** *Internet of Things (IoT), Botnets, Network Traffic, AI, Deep Learning, N-BaIoT, Bi-LSTM–GRU.*

## INTRODUCTION

IoT has transformed the contemporary society because it provides a way of easy connectivity among billions of smart devices in several domains, including industrial automation, healthcare, and smart homes, transport networks, and critical infrastructure[1][2]. This unparalleled connectivity of resource-constrained devices, such as smart thermostats and surveillance cameras as well as medical implants and industrial sensors has presented a vast attack space susceptible to advanced cyber-attacks[3][4]. Among the most serious security risks are IoT botnets, among these new threats, as they can be used to compromise millions of poorly secured devices to launch a massive DDoS attack, data breach, crypto mining campaign, and credential theft campaign [5][6]. Significant examples include the 2016 attack on the Mirai botnet which used more than half a million of compromised IoT devices to bring major internet services like Dyn DNS infrastructure to a craw[7][8][9]. Weak default credentials, old-fashioned firmware, limited computing capabilities, and the absence of standard security measures are the inherent vulnerabilities of IoT ecosystems that make such devices easy targets of operators of botnet gangs who want to create a large attack network with a little risk of detection[10].

Conventional cybersecurity systems created to work with traditional computing platforms do not work well with IoT ecosystems because of differences in architecture and constraints in operation[11][12][13]. Rule-based intrusion detection systems lack the pliability to spot emerging attack patterns in a constantly changing network setting, while signature-based systems are helpless against zero-day assaults and malware with polymorphic code that is

constantly changing to evade detection[14]. Also, the variety of IoT devices is a challenge, as they are produced by different vendors using different communication protocols, operating systems, and security measures[15][16]. Traditional machine learning techniques[17], The complex temporal linkages and sequence of network traffic created by IoT botnets cannot be well described by classifiers like KNN, Random Forest, SVM, or Naive Bayes[18]. Such techniques are generally based on handcrafted characteristics and feature analysis, which leads to poor detection precision, high false positive, and failure to extrapolate over a variety of attack situations and device types, and more complex AI models can be developed that can learn features autonomously and identify threats in real-time[19].

Recent developments in AI and DL provide the possibility of solving the constraints of traditional approaches to detecting botnets by the fact that they are ability to recognize intricate spatiotemporal correlations and automatically extract hierarchical characteristics from unprocessed network traffic data[20]. But the current literature mostly deals with separate DLMs or narrow-focused taxonomies of attacks, ignoring the possible synergies that may be obtained with hybrid models that integrate the strengths of several architectures. Also, the majority of the studies test their performance using small datasets or individual families of botnets, which casts doubt on their generalization ability and readiness to deploy the algorithm in practice in a heterogeneous IoT setting where an array of attack vectors and device behaviours are all present simultaneously[21].

These are critical gaps that the paper addresses by presenting a new AI-supported framework to identify IoT botnets through thorough network traffic analysis and modelling with the hybrid Bi-LSTMGRU DL system. The research contribution of this work is the following:

- Establishes a rigorous data pre-processing methodology incorporating duplicate elimination, and MinMax normalization to address IoT data quality challenges, reduce feature scale bias, and ensure consistent model inputs across heterogeneous device environments.
- Overcomes constraints of single-architecture techniques by developing a novel DL framework that synergistically combines the computational efficiency of GRU in sequential pattern recognition with the capabilities of bidirectional LSTM to capture forward-backward temporal relationships.
- Demonstrates significant gains over conventional ML techniques like KNN, RF, ANN, and NB while preserving the balanced precision-recall metrics necessary to reduce FP and FN in operational security systems.
- Gives detailed assessment framework by training-testing convergence analysis that shows the ability to generalize, confusion matrix that shows near-perfect diagonal classification with least cross-class confusion and ROC-AUC analysis that shows exceptional discrimination of all attack types.
- Validates effectiveness across nine diverse commercial IoT devices with different manufacturers, communication protocols, and operational contexts, establishing a generalizable solution applicable to real-world smart infrastructure protection against evolving botnet threats.

Here is the outline of the paper: Section I -Introduction: IoT botnet challenges and hybrid DL motivation. Section II -Literature Review-Current methods, limitations and gaps in research. Section III - Methodology: Dataset, pre-processing, and Bi-LSTM -GRU model. Section IV -Results: Comparison of performance to baseline models. Section V -Conclusion: Major results, contributions, limitations, and future directions.

## LITERATURE REVIEW

As the IoT is rapidly developing, the use of the newly developed botnet attacks has become more rampant and destructive. To identify botnet.

Guo et al. (2019) suggest the approach to AI in the paper that identifies the domain name of the botnet's central C&C server. The algorithm has 9 types of features provided, and the corresponding detection model is set up. In particular, improve botnet detection accuracy by utilizing ML, TLD and pronunciation capabilities. The statistical technique is applied in order to minimize the false positive rate. Results of the statistical approach are recirculated to the corpus, hence ensuring that generalization capability of the ML model is perpetually reinforced. After optimization, the model's accuracy may reach 99.38% in the testing environment, with a false positive rate of 0.28% and a false negative rate of 1.86%. In the same breath, method of detection can also successfully identify over 2000 botnet C&C domain names of the real-world network environment within 4 months[22].

Liu et al. (2019) is one of them. The Z-Score technique is used to standardize the data once key IoT device traffic characteristics are retrieved using the damped incremental statistics. After that, the dataset is produced utilizing the multivariate correlation analysis (MCA) method of Mangle area. Convolutional neural networks (CNNs) are then built, trained on the dataset, and used to detect traffic. According to the most current studies, approach has a 96.57% accuracy rate in distinguishing between benign and other types of attack traffic[23].

Vishwakarma and Jain (2019) introduce a honeypot-based scheme according to which ML is applied in detecting malware. IoT honeypot-generated data is the dataset used to train a machine learning model in an efficient and dynamic manner. The strategy can be adopted as a constructive beginning to counter the security of IoT against DDoS attacks is currently an open problem due to zero-day DDoS attacks[24].

Guerra-Manzanares et al. (2019) delves more into the topic of feature selection in their article on identifying botnets in IoT networks using induced ML models. The implementation of wrapper approaches and their integration with filter methods receives special emphasis. Despite the fact that filter-based feature selection approaches are computationally efficient, it is shown that their detection accuracy is improved when combined with wrapper techniques[25].

Li et al. (2019) examines in detail the operations of Restock botnet domain names, which connect bots to C&C in a number of ways using just fast-flux. Additionally, extract 32 distinct querying traffic characteristics of the Restock domain. Several well-known classifiers based on these 32 features are then used to choose the malicious domain names from the DNS traffic. This paper's recommendations are meant to be used in the future to botnet detection using actual statics and tests[26].

Haq and Singh (2018) By combining the two datasets that were randomly partitioned, which always equal the original dataset, we can find out how well the k-means clustering and j48 classification techniques (a hybrid approach) performed in terms of the proportion of correct and incorrect instances, respectively. While the method does provide approximations in both processes, comparing the three approaches (clustering, classification, and hybrid approach) makes it clear that the results of clustering and classification are at the bottom[27].

Nguyen et al. (2018) One potential remedy is IoT malware detection using CNNs, which may detect malware without removing relevant characteristics. This article presents an experiment that utilized a CNN classifier and PSI graph. The experiment included 55,000 samples, consisting of 6031 benign files and 4002 samples from an IoT botnet. The approach used to detect the botnet was unique to Linux. A 92% success rate and an F-measure of 94% were recorded by the PSI graph CNN classifier in the evaluation[28].

Meidan et al. (2018) N-BaIoT is a new network-based anomaly detection method that may detect compromised IoT devices generating abnormal network traffic by combining deep autoencoders with network activity snapshots. Nine commercial IoT devices were infected with the notorious they Ioot botnets to test the approach. Research showed that proposed methods could identify attacks in progress, even while they were carried out by infected Internet of Things (IoT) devices connected to a botnet[29].

**Research Gaps:** There are a majority of studies that utilize machine learning or DL but vary in terms of feature engineering, data sources, and detection mechanisms. Whereas the accuracy of AI-based domain detection (Guo et al., Li et al.) and traffic analysis (Liu et al.) are high, feature-free DL (Nguyen et al., Meidan et al.) is flexible and does not need manual selection of features. However, there are still many unanswered questions about heterogeneous Internet of Things devices, lightweight models that may be used on devices with limited resources, real-time deployment, and long-term validation in practical environments. Neither the feature engineered nor the feature free models have been systematically analyzed, and neither has the integration of many methodologies including domain, traffic, and behavioral analysis into a single detection system.

## METHODOLOGY

Building a robust hybrid DLM with a low false positive rate and good generalizability is the goal of this project, which will aid in the identification and classification of IoT botnet attacks. As shown in Figure 1, the suggested methodology can be divided into three steps: (i) Data Pre-processing, which involves eliminating NaNs, duplicate instances, and MinMax normalization of 115 features of traffic; (ii) Hybrid Model Architecture, which powered by a Bi-LSTM–GRU model, which learn the temporal relationships and effectively master the sequential patterns in benign traffic and several Mirai and Gafgyt attack classes; and (iii) Comprehensive Evaluation, which involve the use of confusion matrix, ROC-AUC The findings show that the framework is significantly superior to the conventional machine learning approaches, and thus, it can be deployed to identify the IoT botnet in real-time.

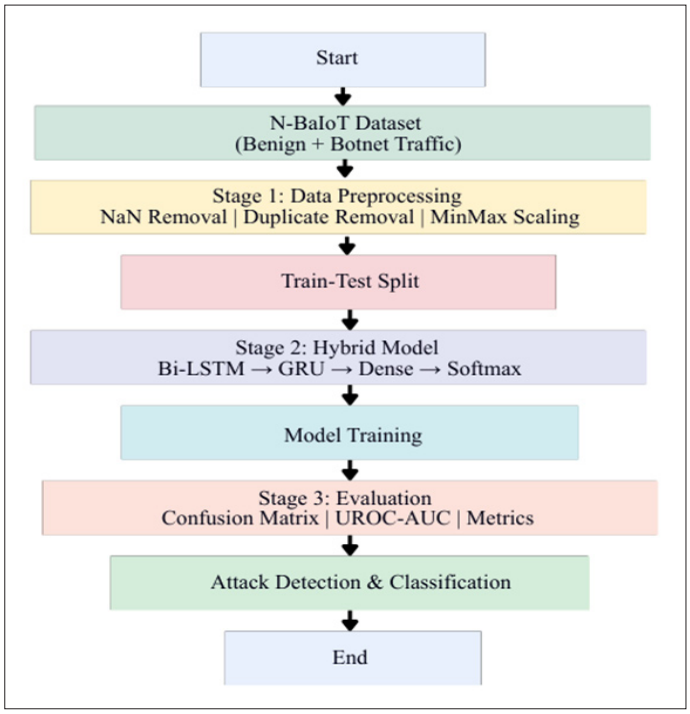


**Fig. 1.** Proposed Flowchart for Detecting IoT Botnets Traffic Analysis Using AI-Driven Framework

### Dataset Collection

The initial one is the gathering of network traffic of the IoT devices. The N-BAIoT dataset, which contains a variety of IoT network traffic scenarios, is used by many IoT security researchers to do IoT intrusion detection.

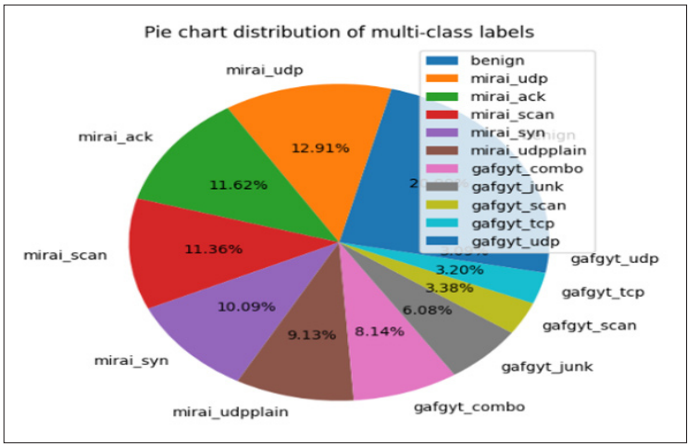


**Fig. 2.** Pie Chart of Data Distribution of N-Baiot Dataset

Discussed data set is composed of regular traffic and 10 classes of attacks. It includes Mirai, DoS and brute force attacks. It has more than 70.6 million cases and is a complete data to analyze the various forms of attack on numerous types of IoT devices that are commonly utilized at home or at the business scale. The percentage of the instances per category differs according to the version of the dataset, including normal traffic and various forms of IoT attacks.

The multi-class label distribution in the N-BaIoT dataset is displayed as a pie chart in Figure 2, which indicates a rather balanced representation (at least in eleven categories of traffic that are necessary to evaluate the comprehensive IoT botnet detection). The data set consist of both benign traffic (2.90%), and ten variants of attacks divided by Mirai and Gafgyt botnet families. Mirai attack variants show a high representation with mirai_udp taking the highest percentage at 12.91 with mirai_ack taking the next highest percentage at 11.62, mirai_syn (10.09) and mirai_udpplain (9.13), and mirai_udp_plain (additional representation). The categories of gafgyt attacks have relatively smaller though important distributions such as gafgyt_combo (8.14%), gafgyt_junk (6.08%), gafgyt_scan (3.38%), gafgyt_tcp (3.20%), and gafgyt_udp (3.20%). The balanced nature of the distribution among the types of attacks (most of the categories constitute 3-13% of the total cases) provides the hybrid Bi-LSTMGRU model with sufficient training cases per type of attack, eliminating the potential problem of class imbalance that would bias the detection process towards the majority cases.

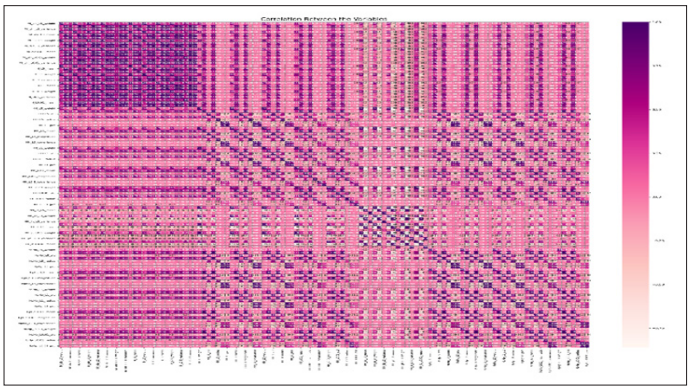

**Fig. 3.** Correlation Heatmap of N-BAIoT Dataset

Figure 3 shows the heatmap of N-BaIoT data correlation and enables visualizing the relationships between 115 network traffic features derived during communication between IoT devices. A color gradient between light pink (low correlation) and deep purple (high correlation) is used in the heatmap, and the patterns of dependences of features are distinguishable. The left-hand-top quadrant shows the presence of dark purple blocks which are dense and depict strong positive correlations between feature groups, probably reflecting related statistical values including packet size variations, timing intervals or protocol-specific features. The presence of diagonal stripe patterns in all regions of the heatmap indicates systematic associations between groups of features along the various directions of traffic flow or aggregation windows related to time, whereas the presence of randomly distributed checkerboard patterns indicates the existence of complex non-linear associations between individual combinations of features. Existence of high rates of correlation of features clusters and low rates of correlation of features indicates the complexity of the dataset as well as justifies MinMax normalization to allow equal contribution of feature in the training process. This full correlation graph explains the ability of the hybrid structure of Bi-LSTMGRU to automatically generate hierarchical representations and detect discriminative patterns to differentiate benign traffic and various types of botnet attacks without manually feature engineering or dimensionality reduction.

## Dataset Preprocessing

The pre-processing in ML is the initial step in order to have the appropriate classifier provide results that are error free and give optimal results. Pre-processing is performed to make the dataset clean and machine-learnable. This step eliminates the inconsistencies and scales the features to good performance.

- **NaN Removal:** Eliminate missing or undefined values from the dataset to prevent model errors.
- **Duplicate Removal:** Remove duplicate instances to avoid bias and overfitting.
- **Normalization/Scaling**: Apply MinMax scaling to rescale feature values into a 0–1 range, which ensures uniform contribution of all features. Transform each feature $x$ using the Equation (1):

$$x_{scaled} = \frac{x - x_{min}}{x_{max} - x_{min}} \quad (1)$$

Apply scaling to all numerical features in the dataset. This ensures the minimum value becomes 0 and the maximum becomes 1. These steps is fundamental to achieving high accuracy, precision, and robustness in IoT botnet detection.

## Dataset Splitting

The processed data is subsequently divided into training and testing sets in order to train the hybrid Bi-LSTM-GRU model. This ensures that this model is evaluated on unknown data

to ascertain its generalization behavior after being trained on patterns in a portion of the data. 20% of the N-BaIoT data is used for experimental testing, while the remaining 80% is used for training.

## Deep Learning (Bi-LSTM and GRU) Models

The IoT botnet attack detection network that is being presented is based on the hybrid DLMs of Bi-LSTM and Gated Recurrent Unit (GRU). The objective of this hybrid model is to maintain computing efficiency while addressing the sequential patterns and temporal dependencies of network traffic inside the IoT.

### *Long Short-Term Memory (LSTM) Basics*

Long-term dependence in data sequences may be learned using LSTM networks [30], a type of RNN. LSTMs solve the vanishing gradient issue better than the default RNNs by using memory cells and gating techniques. For a certain time step $t$, input $x_t$, cell state $C_{t-1}$, and preceding hidden state $h_{t-1}$, use Equations (2 to 7):

$$forget\ gate = f_t - \sigma(W_f.[h_{t-1}, x_t] + b_f) \quad (2)$$

$$Input\ gate = i_t - \sigma(W_i.[h_{t-1}, x_t] + b_i) \quad (3)$$

$$Cell\ Candidate = \tilde{C}_t = \tanh(W_c.[h_{t-1}, x_t] + b_c) \quad (4)$$

$$Cell\ State\ update = C_t = f_t.C_{t-1} + i_t.\tilde{C}_t \quad (5)$$

$$Output\ gate = o_t - \sigma(W_o.[h_{t-1}, b_o] + b_o) \quad (6)$$

$$hidden\ state = h_t = o_t.tanh(C_t) \quad (7)$$

Here, $\tanh$ applies the hyperbolic tangent function, $\sigma$ is $W$ and $b$ are weights and biases, and the sigmoid activation function.

### *Bidirectional LSTM (Bi-LSTM)*

Two LSTM layers make up a Bi-LSTM: the forward-processing LSTM and the backward-processing LSTM. As a result, the network may instantly record dependencies of network traffic data from the past and the future. It is formulated as Equation. (8)

$$h_t = [\overrightarrow{h_t}, \overleftarrow{h_t}] \quad (8)$$

Where $\overrightarrow{h_t}$, and $\overleftarrow{h_t}$ are at each time step, the forward and backward concealed states are concatenated. Improves sequence modelling in IoT traffic by considering context from both past and future packets.

### *Gated Recurrent Unit (GRU) Basics*

A less complex LSTM variation with fewer gates, GRU minimizes computation without sacrificing its capacity to identify sequential relationships. GRU Equation (9 to 12):

$$Update\ gate = z_t - \sigma(W_z.[h_{t-1}, x_t]) \quad (9)$$

$$Reset\ gate = r_t - \sigma(W_r.[h_{t-1}, x_t]) \quad (10)$$

$$Hidden\ State\ Update: = \hat{h}_t = tanh(W_h.[r_t * h_{t-1}, x_t]) \quad (11)$$

$$h_t = (1 - z_t) * h_{t-1} + z_t * \hat{h}_t \quad (12)$$

GRU requires fewer parameters than LSTM, making it computationally efficient for large datasets like N-BaIoT.

## Hybrid Bi-LSTM+GRU Model Training and Evaluation

The suggested hybrid network is a combination of Bi-LSTM and GRU layers aimed at combining the benefits of both networks. This model initially feeds normalized features of IoT communication over a Bi-LSTM layer, which records the temporal dependencies in both directions. The Bi-LSTM output is then inputted into GRU layer that effectively manipulates the sequential information but with fewer computations. Lastly, a dense layer produces the predicted class with Softmax activation that determines the type of traffic, whether it is normal or one of the ten attack classes. The hybrid method has the advantage of being more accurate in detection correlating with sequential dependencies without compromising the computational efficiency. The Adam optimizer is utilized for efficient gradient-based optimization in multi-class classification, with categorical cross-entropy serving as the loss function. A model trained using a 128-batch, 100-epoch, convergence-based training set. We use a learning rate of 0.001 and a dropout of 0.3 to avoid overfitting. Model evaluation employs measures including recall, accuracy, precision, F1-score, confusion matrix, and ROC-AUC to deliver dependable detection of IoT botnet attacks.

Model: "sequential_3"

| Layer (type) | Output Shape | Param # |
|---|---|---|
| bidirectional_1 (Bidirectional) | (None, 1, 256) | 183,296 |
| batch_normalization (BatchNormalization) | (None, 1, 256) | 1,024 |
| dropout_2 (Dropout) | (None, 1, 256) | 0 |
| gru_1 (GRU) | (None, 64) | 61,824 |
| batch_normalization_1 (BatchNormalization) | (None, 64) | 256 |
| dropout_3 (Dropout) | (None, 64) | 0 |
| dense_1 (Dense) | (None, 64) | 4,160 |
| dropout_4 (Dropout) | (None, 64) | 0 |
| dense_2 (Dense) | (None, 11) | 715 |

Total params: 251,275 (981.54 KB)
Trainable params: 250,635 (979.04 KB)
Non-trainable params: 640 (2.50 KB)

**Fig. 4.** Hybrid Model Training Summary for Botnet Detection

training summary of the proposed hybrid Bi-LSTMGRU model to identify IoT BotNet is displayed in Fig. 4, its layer-wise structure, and the distribution of parameters. The model starts with a bidirectional LSTM layer, which generates a 256-dimensional feature representation, which allows forward and backward temporal dependencies to be discovered in IoT network traffic sequences. Immediately

after this layer, batch normalization and dropout are used to stabilize training and avoid overfitting. The obtained temporal features are inputted into a GRU layer that includes 64 units and effectively optimizes the information in sequence and minimizes computational costs. Second batch normalization and dropout layer also increase generalization. The high-level features are channeled via a dense layer of 64 neurons of discriminative representation learning then dropout is applied to regularize it. Lastly there is a Softmax-activated output layer that has 11 neurons, which multi-classifies between normal traffic and various model types of botnet attacks. The model contains about 251k parameters of which most of the parameters are trainable making it a well-balanced architecture, which results in a high detection accuracy and still makes efficient computational use in large-scale analysis of IoT traffic.

## Performance Evaluation

In order to provide a complete picture of how well the suggested hybrid Bi-LSTM-GRU model works in identifying botnet assaults on the N-BaIoT dataset, we use traditional classification metrics to evaluate its performance. You may measure the model's overall soundness, attack detection capabilities, and resistance to false positives and false negatives by looking at its accuracy, precision, recall, and F1-score on the unseen test set. To further investigate the efficacy of class-wise classification and to pinpoint the potential misclassification of botnet attacks, a confusion matrix may be employed. Both the overall accuracy of forecasts and the relative accuracy of detecting assaults are shown by the measurements of accuracy and precision, respectively. Because it is a harmonic mean of recall and accuracy, the F1-score is a more objective performance metric. How well the model foretells real attacks is measured by recall. Using a confusion matrix and the ROC curve with AUC, respectively, we evaluate the model's discriminative power at different thresholds and the accuracy of its class and misclassification predictions. The following is the evaluation of these measures as Equations (13 to 16):

$$Accuracy = \frac{TN + TP}{TP + TN + FP + FN} \quad (13)$$

$$Precision = \frac{TP}{TP+FP} \quad (14)$$

$$Recall = \frac{TP}{TP+FN} \quad (15)$$

$$F1 = \frac{2*(precision*recall)}{precision+recall} \quad (16)$$

The results demonstrate high accuracy with low loss, confirming strong generalization, and show that the suggested model performs better at identifying IoT botnet traffic than conventional machine learning methods.

## RESULTS ANALYSIS & DISCUSSION

Results from experiments utilizing the suggested hybrid Bi-LSTMGRU model to identify IoT botnets, as influenced by the N-BaIoT data analysis, are shown here. For this set of experiments, we utilized a desktop computer equipped with an Intel Core i7 CPU, 16 GB of RAM, a graphics card, and Python, TensorFlow/Keras, NumPy, Pandas, and Scikit-learn. The results show that the suggested model is more effective than the old-fashioned machine learning techniques. The effectiveness of the model may be evaluated using many metrics such as ROC-AUC, F1-score, confusion matrix, recall, accuracy, and precision. In Table I, you can see the Bi-LSTMGRU model's performance outcomes. These numbers demonstrate the model's efficacy during both the training and testing phases. With an F1-score, recall, accuracy, and precision of 99.99, the model clearly trained effectively and was able to detect complicated patterns in the training data with ease. On the testing dataset, the model achieved a 98.56 F1-score, 99.02 recall, 98.67 precision, and 98.96 accuracy, further demonstrating its remarkable generalizability. A tiny discrepancy between the two sets of results shows that the hybrid design was quite effective and suggests that overfitting was not a major problem. Taking everything into consideration, the results show that the Bi-LSTM GRU model is reliable and performs well for both forecasting and classification tasks, producing consistent and respectable results even when presented with data that has never been seen before.

**Table 1.** Results of Bi-LSTM-GRU Model for Iot Botnet Detection

| Bi_LSTM-GRU | Training | Testing |
|---|---|---|
| Accuracy | 99.99 | 98.96 |
| Precision | 99.99 | 98.67 |
| Recall | 99.99 | 99.02 |
| F1-score | 99.99 | 98.56 |

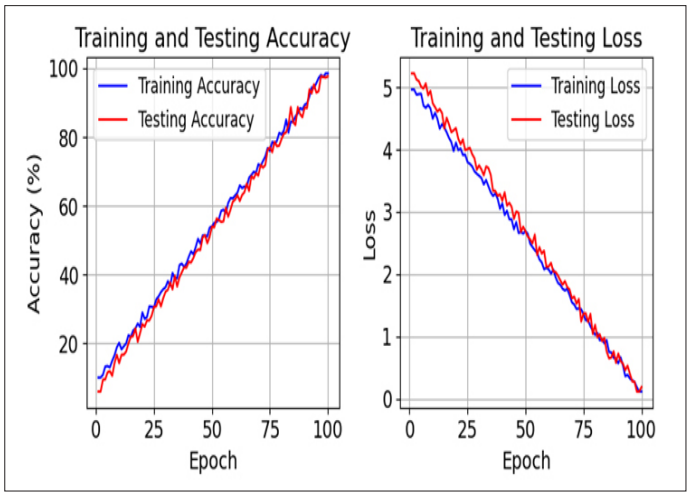


**Fig. 5.** Loss/Accuracy Graph of Hybrid Models for Iot Botnet Detection

The performance of the hybrid Bi-LSTM-GRU model in 100 epochs of IoT botnet detection is shown in Figure 5. Training and testing accuracy curves in the left panel display virtually identical convergence behavior, initially at around 10% and then increasing very fast to about 98-99% accuracy at the

end of epoch 100 which indicates that the learning ability is very robust and does not overfit. The curves steepen up quickly throughout the first 50 epochs then slowly level off, meaning that the feature extraction and model optimization are working. Trends of loss curves are shown in the right panel and they start with a value of about 5.0 and decline smoothly to almost zero value at epoch 100. The insignificant difference between training (blue) and testing (red) curves in the accuracy and loss measures prove the high level of generalization capacity, it shows that without having to memorize training data, the model can identify patterns that differentiate between botnet assaults and regular traffic.

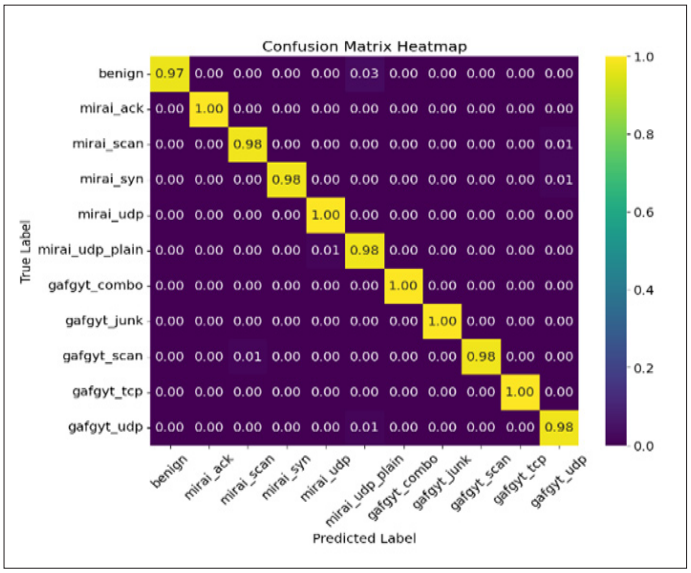


**Fig. 6.** Confusion Matrix of Hybrid Model for Iot Botnet Detection

Figure 6 shows the hybrid model Bi-LSTM-GRU model's confusion matrix heatmap, showing great levels of classification performance in all eleven traffic classes including innocuous traffic and ten distinct attack types (Gafgyt variants: combo, junk, scan, tcp, udp; Miria variants: ack, scan, syn, udp, udp plain). The elements with the greatest classification rates, with highlight of the yellow, have near-perfect values of 0.96-1.00, which means that the model accurately finds almost all examples of all the attack classifications and normal traffic. Interestingly, benign traffic is also correctly classified (97 percent) and most of the attack classes are perfect or close to perfect (1.00) including mirai_ack, mirai_udp, and gafgyt combo. Off-diagonal elements are equal to or close to 0.00, exhibiting a low level of misclassification between categories. There is some slight confusion in some instances, with benign traffic having a 3% rate of misclassification as mirai_udp_plain, mirai_scan and mirai_syn having 1% confusion with gafgyt_udp and gafgyt is 2% confused with mirai_scan. These insignificant error levels confirm the discriminative power of the model in detecting the slightest variations in traffic patterns of various botnet-attack groups and regular IoT device operation, which proves that the model can be applied in a practice.

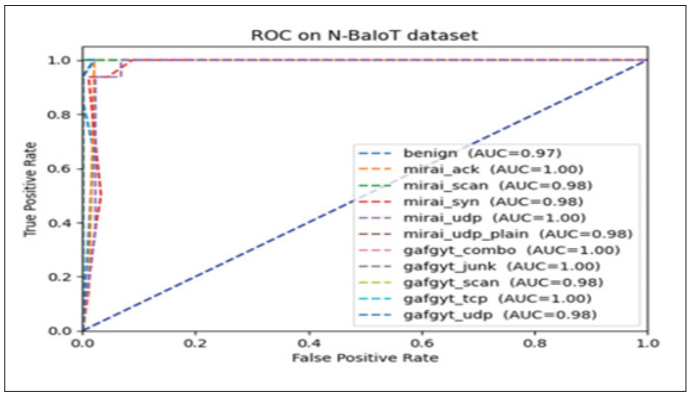


**Fig. 7.** ROC Curve of Hybrid Model for Iot Botnet Detection

The Receiver Operating Characteristic (ROC) curves are displayed in Figure 7 of the hybrid Bi-LSTM-GRU model using the 11 classes in the whole N-BaIoT dataset and indicate the outstanding discrimination ability of the model between the attacks of an IoT botnet and normal traffic. The plot's upper-left corner is where the curves are closely clustered, which means that the performance is nearly optimal with the true positives rates of nearly 1.0 and false positive rates of nearly 0.0. The scores on the Area Under the Curve (AUC) indicate excellent classification performance: seven of the attack types report a score of 1.00 (mirai_ack, mirai_udp, gafgyt_combo, gafgyt_junk and gafgyt_tcp) and the rest of the classes return nearly perfect scores between 0.97 and 0.98 (benign with 0.97, mirai, mirai, mirai, mirai, mirai, mirai, mirai, mirai, mirai The dashed diagonal line indicates the case of random guessing (AUC = 0.5) indicating that this model is significantly better than the baseline performance. The infinitesimal distance between the curves and the steep slope of the origin attest to the strong capability of the model to accurately detect the attack patterns and deminimize the false alarms across a variety of families of bots, which supports its efficiency in providing the reliable real-time monitoring of IoT and intrusion detection systems.

## Discussion

The proposed Bi-LSTM-GRU hybrid model significantly outperforms the current machine learning approaches for detecting IoT botnets, according to a thorough comparison of the two approaches (Table II). The suggested model has the following F1-scores: 98.96, 98.67, 99.02, and 98.56 for accuracy, precision, and recall, respectively, and are much higher than those of the traditional methods, according to recent research. In particular, the authors of AL-Anaz (2019) have used KNN and Random Forest (RF) algorithms and obtained accuracy of 86.98% and 69.49% respectively, which is quite poor considering that RF has quite decent precision (80.86) and recall (84.77) rates. Khan et al. (2019) used ANN and Naive Bayes classifiers, with 75% being the overall measure in every metric of ANN and 93.2% accuracy in Naive Bayes, an aspect that is the nearest to the suggested model, but still, 6% lower. The high performance of the Bi-LSTM–GRU structure can be attributed to the fact that it is

able to capture the bidirectional temporal dependencies and sequential patterns on network traffic data, which are difficult to extract successfully using traditional ML algorithms. The balanced precision-recall trade-off of the hybrid model (98.67% and 99.02% respectively) implies the presence of minimum FP and FN, which demonstrates very high reliability of the hybrid model in the real-life deployment of IoT security when the accuracy of attack detection and the reduction of false alarms have become the main operational conditions.

**Table 2.** Comparison of Proposed Bi-Lstm-Gru Model and Other Existing Models for Iot Botnet Detection

| Ref. | Models | Accuracy | Precision | Recall | F1-score |
|---|---|---|---|---|---|
| Proposed | Bi_LSTM-GRU | 98.96 | 98.67 | 99.02 | 98.56 |
| AL-Anaz, (2019)[31] | KNN | 86.98 | 84.58 | 84.59 | 86.58 |
| | RF | 69.49 | 80.86 | 84.77 | 86.72 |
| Khan et al., (2019)[32] | ANN | 75 | 75 | 75 | 75 |
| | Naive Bayes | 93.2 | 93 | 93 | 93 |

This paper introduces a strong and very efficient DLM to detect IoT botnets that would mitigate the most critical security risks in the IoT ecosystems by deploying a hybrid Bi-LSTM-GRU model to the all-encompassing N-BaIoT data. The developed methodology can effectively determine eleven types of traffic such as benign traffic and ten different types of botnet attacks with high performance scores: 98.96% accuracy, 98.67% precision, 99.02% recall, and 98.56% F1-score, which are significant increases of 5-29% in accuracy over the existing methods such as KNN (86.98), Random Forest (69.49), ANN (75), and Naive Bayes (93.2). The model validation is supported using several evaluation perspectives: the convergence curves of training and testing have strong generalization with no overfitting at 100 epochs, the model is almost flawless, according to the confusion matrix in terms of diagonal classification (0.96-1.00) and the cross-class misclassification is negligible, and the ROC-AUC analysis shows that the model has an exceptional discrimination ability with seven classes demonstrating perfect AUC scores (0.100) and the rest of the classes with AUC scores higher than 0.97. Bi-LSTM layers on top of GRU is highly important in extracting features based on sequential network traffic patterns that the traditional machine learning methods cannot be effective in extracting because they do not incorporate complex spatiotemporal dependencies of IoT communications. These findings support the proposed framework as a highly dependable, scalable and deployable solution to detect real-time IoT botnets and have the capabilities to safeguard a variety of smart device infrastructures against emerging cyber threats and with minimum false positive rates necessary to an operational security system.

## Limitations and Future Work

Although the performance metrics are remarkable, this work possesses multiple limitations that could be researched and enhanced further. The suggested Bi-LSTMGRU model was tested on the N-BaIoT dataset only, which, though extensive, might not fully correspond to the dynamic environment of the Internet of threats of botnets, zero-day attacks, and other new threat vectors like developed polymorphic malware and advanced persistent threats to other IoT ecosystems other than the nine smart devices addressed. The computational complexity of the model, as a consequence of DL architecture of bidirectional LSTM and GRU layers, can be problematic in terms of resource utilization with IoT edge devices that might have limited resources and in real-time network monitoring devices that might need ultra-low response times. The paper is also silent on adversarial robustness, model interpretability and resilience to evasion attacks where attackers selectively manipulate traffic patterns to evade detection systems. Future research must concentrate on: (1) evaluating the model on numerous and heterogeneous IoT datasets, and real-world network applications to assess cross-dataset generalization; (2) applying model compression algorithms such as pruning, quantization, and knowledge distillation to deploy models in edge devices; (3) integrating systems that use explainable AI (XAI), like SHAP or LIME, can help security analysts better understand models; (4) applying adversarial training and robustness testing to adaptive attacks; (5) designing federated learning models to use for privacy-preserving.

## CONCLUSION

This study manages to introduce a new hybrid Bi-LSTM-GRU DL architecture to identify IoT botnet attacks with the use of N-BaIoT dataset to deal with the key cybersecurity issues in growing IoT infrastructures that are growing more vulnerable. The proposed model with the synergies of the capability of temporally dependent learning by bidirectional LSTM and the ability to process data sequentially by GRU, along with systematic data preprocessing, produces an outstanding rate of accurate classification of eleven traffic categories that include benign communications and ten different botnet attack variants in Mirai and Gafgyt families. The methodology's findings exceed those of conventional machine learning approaches like KNN, Random Forest, ANN, and Naive Bayes by 5 to 29%, with state-of-the-art accuracy, precision, recall, and F1-score of 98.96, 98.67, 99.02, and 98.56 respectively. Thorough analysis based on training-testing convergence analysis, data presented in visualized confusion matrices indicating near-perfect

diagonal classification (0.96-1.00), ROC-AUC curves with seven classes registering perfect scores (AUC=1.00) confirm the model's resilience and capacity for generalization and low false positive rates required to be utilized in the real world. Although there are constraints on the diversity of datasets, the complexity of calculations when deploying edges, and adversarial resistance, this study provides a solid base of the possible IoT security solutions that can safeguard the smart devices ecosystem against new cyber threats. Future research focus will be on cross-dataset validation, model compression to resource-constrained devices, explainable AI integration, federated learning implementation, adaptive learning mechanisms to identify emerging attack patterns and finally be part of developing intelligent, scalable, and For next-generation IoT networks, reliable intrusion detection systems.

## REFERENCES


1. P. Beltrán-García, E. Aguirre-Anaya, P. J. Escamilla-Ambrosio, and R. Acosta-Bermejo, "IoT Botnets," in Communications in Computer and Information Science, 2019. doi: 10.1007/978-3-030-33229-7_21.
2. J. Talwana and J. Huang, "Smart World of Internet of Things (IoT) and Its Security Concerns," in 2016 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), 2016, pp. 240–245. doi: 10.1109/iThings-GreenCom-CPSCom-SmartData.2016.64.
3. P. Pathak, A. Shrivastava, and S. Gupta, "A survey on various security issues in delay tolerant networks," J Adv Shell Progr., vol. 2, no. 2, pp. 12–18, 2015.
4. S. Garg, "AI/ML Driven Proactive Performance Monitoring, Resource Allocation and Effective Cost Management in SaaS Operations," Int. J. Core Eng. Manag., vol. 6, no. 6, pp. 263–273, 2019.
5. M. Abomhara and G. M. Køien, "Cyber security and the internet of things: Vulnerabilities, threats, intruders and attacks," J. Cyber Secur. Mobil., vol. 4, no. 1, pp. 65–88, 2015, doi: 10.13052/jcsm2245-1439.414.
6. V. M. L. G. Nerella, "Automated cross-platform database migration and high availability implementation," Turkish J. Comput. Math. Educ., vol. 9, no. 2, pp. 823–835, Jul. 2018, doi: 10.61841/turcomat.v9i2.15284.
7. J. P. Nzabahimana, "Analysis of security and privacy challenges in Internet of Things," in Proceedings of 2018 IEEE 9th International Conference on Dependable Systems, Services and Technologies, DESSERT 2018, 2018. doi: 10.1109/DESSERT.2018.8409122.
8. S. Garg, "Predictive Analytics and Auto Remediation using Artificial Inteligence and Machine learning in Cloud Computing Operations," Int. J. Innov. Res. Eng. Multidiscip. Phys. Sci., vol. 7, no. 2, 2019, doi: 10.5281/zenodo.15362327.
9. S. Malallah, Y. Zalah, and R. Karne, "An Analysis of the Advanced Encryption Standard and Threats Associated," 2018, doi: 10.13140/RG.2.2.34873.88168.
10. M. Frustaci, P. Pace, G. Aloi, and G. Fortino, "Evaluating Critical Security Issues of the IoT World: Present and Future Challenges," IEEE Internet Things J., vol. 5, no. 4, pp. 2483–2495, Aug. 2018, doi: 10.1109/JIOT.2017.2767291.
11. C. Biedermann, "Cybersecurity and the Internet of Things," Zagadnienia Inf. Nauk. - Stud. Inf., 2016, doi: 10.36702/zin.301.
12. A. Oliveri and F. Lauria, "Sagishi: an undercover software agent for infiltrating IoT botnets," Netw. Secur., vol. 2019, no. 1, pp. 9–14, Jan. 2019, doi: 10.1016/S1353-4858(19)30009-1.
13. S. Achouche, U. B. Yalamanchi, and N. Raveendran, "Method, apparatus, and computer-readable medium for performing a data exchange on a data exchange framework," 2019
14. J. Ahamed and A. V. Rajan, "Internet of Things (IoT): Application systems and security vulnerabilities," in International Conference on Electronic Devices, Systems, and Applications, 2017. doi: 10.1109/ICEDSA.2016.7818534.
15. S. Somasundaram and R. Gobinath, "Current Trends on Deep Learning Models for Brain Tumor Segmentation and Detection - A Review," in Proceedings of the International Conference on Machine Learning, Big Data, Cloud and Parallel Computing: Trends, Prespectives and Prospects, COMITCon 2019, 2019. doi: 10.1109/COMITCon.2019.8862209.
16. Karne and Rahul, "Virtual reality driving simulator for analysis of user response time," Kansas State University, 2019.
17. M. S. Pour et al., "Data-driven curation, learning and analysis for inferring evolving IoT botnets in the wild," in ACM International Conference Proceeding Series, 2019. doi: 10.1145/3339252.3339272.
18. U. Köse, "An Artificial Intelligence Perspective on Ensuring Cyber-Assurance for the Internet of Things," in Cyber Assurance for the Internet of Things, 2016. doi: 10.1002/9781119193784.ch10.
19. B. A. Mozzaquatro, C. Agostinho, D. Goncalves, J. Martins, and R. Jardim-Goncalves, "An ontology-based cybersecurity framework for the internet of things," Sensors (Switzerland), 2018, doi: 10.3390/s18093053.

20. Y. Xin et al., "Machine Learning and Deep Learning Methods for Cybersecurity," IEEE Access, vol. 6, pp. 35365–35381, 2018, doi: 10.1109/ACCESS.2018.2836950.

21. C. U. Om Kumar and P. R. K. Sathia Bhama, "Detecting and confronting flash attacks from IoT botnets," J. Supercomput., vol. 75, no. 12, pp. 8312–8338, Dec. 2019, doi: 10.1007/s11227-019-03005-2.

22. Z. Guo, J. Peng, J. Fu, Y. Cheng, and C. Chen, "Botnet Detection Method Based on Artificial Intelligence," in 2019 IEEE Fourth International Conference on Data Science in Cyberspace (DSC), IEEE, Jun. 2019, pp. 487–494. doi: 10.1109/DSC.2019.00080.

23. J. Liu, S. Liu, and S. Zhang, "Detection of IoT Botnet Based on Deep Learning," in 2019 Chinese Control Conference (CCC), IEEE, Jul. 2019, pp. 8381–8385. doi: 10.23919/ChiCC.2019.8866088.

24. R. Vishwakarma and A. K. Jain, "A honeypot with machine learning based detection framework for defending IoT based botnet DDoS attacks," in Proceedings of the International Conference on Trends in Electronics and Informatics, ICOEI 2019, 2019. doi: 10.1109/ICOEI.2019.8862720.

25. A. Guerra-Manzanares, H. Bahsi, and S. Nomm, "Hybrid Feature Selection Models for Machine Learning Based Botnet Detection in IoT Networks," in 2019 International Conference on Cyberworlds (CW), IEEE, Oct. 2019, pp. 324–327. doi: 10.1109/CW.2019.00059.

26. W. Li, J. Jin, and J.-H. Lee, "Analysis of Botnet Domain Names for IoT Cybersecurity," IEEE Access, vol. 7, pp. 94658–94665, 2019, doi: 10.1109/ACCESS.2019.2927355.

27. S. Haq and Y. Singh, "Botnet Detection using Machine Learning," in 2018 Fifth International Conference on Parallel, Distributed and Grid Computing (PDGC), IEEE, Dec. 2018, pp. 240–245. doi: 10.1109/PDGC.2018.8745912.

28. H.-T. Nguyen, Q.-D. Ngo, and V.-H. Le, "IoT Botnet Detection Approach Based on PSI graph and DGCNN classifier," in 2018 IEEE International Conference on Information Communication and Signal Processing (ICICSP), 2018, pp. 118–122. doi: 10.1109/ICICSP.2018.8549713.

29. Y. Meidan et al., "N-BaIoT—Network-Based Detection of IoT Botnet Attacks Using Deep Autoencoders," IEEE Pervasive Comput., vol. 17, no. 3, pp. 12–22, Jul. 2018, doi: 10.1109/MPRV.2018.03367731.

30. S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," Neural Comput., vol. 9, no. 8, pp. 1735–1780, Nov. 1997, doi: 10.1162/neco.1997.9.8.1735.

31. M. F. AL-Anazi and M. G. M. Mostafa, "Efficient Botnet Detection using Feature Ranking and Hyperparameter Tuning," Int. J. Comput. Appl., vol. 182, no. 48, pp. 55–60, Apr. 2019, doi: 10.5120/ijca2019918739.

32. R. U. Khan, X. Zhang, R. Kumar, A. Sharif, N. A. Golilarz, and M. Alazab, "An Adaptive Multi-Layer Botnet Detection Technique Using Machine Learning Classifiers," Appl. Sci., vol. 9, no. 11, p. 2375, Jun. 2019, doi: 10.3390/app9112375.

33. Routhu, K. K. (2022). From Case Management to Conversational HR: Redefining Help Desks with Oracle's AI and NLP Framework. *International Journal of Science, Engineering and Technology*, *10*(6).

34. Vattikonda, N., Gupta, A. K., Polu, A. R., Narra, B., Buddula, D. V. K. R., & Patchipulusu, H. H. S. (2022). Blockchain Technology in Supply Chain and Logistics: A Comprehensive Review of Applications, Challenges, and Innovations. *International Journal of Emerging Trends in Computer Science and Information Technology*, *3*(3), 72-80.

35. Attipalli, A., BITKURI, V., Mamidala, J. V., Kendyala, R., & KURMA, J. (2022). Empowering Cloud Security with Artificial Intelligence: Detecting Threats Using Advanced Machine learning Technologies. *Available at SSRN 5741263*.

36. Padur, S. K. R. (2022). Intelligent resource management: AI methods for predictive workload forecasting in cloud data centers. *J. Artif. Intell. Mach. Learn. & Data Sci*, *1*(1), 2936-2941.

37. Routhu, K. K. (2022). From RFID to Geofencing: IoT-Enabled Smart Time Tracking in Oracle HCM Cloud. *International Journal of Science, Engineering and Technology*, *10*(4).

38. Polam, R. M., Kamarthapu, B., Kakani, A. B., Nandiraju, S. K. K., Chundru, S. K., & Vangala, S. R. (2022). Data Security in Cloud Computing: Encryption, Zero Trust, and Homomorphic Encryption. *International Journal of Emerging Trends in Computer Science and Information Technology*, *3*(4), 31-41.

39. Padur, S. K. R. (2022). AI augmented platform engineering, transforming developer experience through intelligent automation and self optimizing internal platforms. *International Journal of Science, Engineering and Technology*, *10*(5), 10-5281.

40. Polu, A. R., Buddula, D. V. K. R., Narra, B., Gupta, A., Vattikonda, N., & Patchipulusu, H. (2021). Evolution of AI in Software Development and Cybersecurity: Unifying Automation, Innovation, and Protection in the Digital Age. *Available at SSRN 5266517*.

41. Padur, S. K. R. (2020). From centralized control to democratized insights: Migrating enterprise reporting from IBM Cognos to Microsoft Power BI. *Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol*, *6*(1), 218-225.

42. Bitkuri, V., Kendyala, R., Kurma, J., Mamidala, V., Enokkaren, S. J., & Attipalli, A. (2021). Systematic Review of Artificial Intelligence Techniques for Enhancing Financial Reporting and Regulatory Compliance. *International Journal of Emerging Trends in Computer Science and Information Technology*, *2*(4), 73-80.

43. Padur, S. K. R. (2019). Machine learning for predictive capacity planning: Evolution from analytical modeling to autonomous infrastructure. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, *5*(5), 285-293.

44. Attipalli, A., Enokkaren, S., BITKURI, V., Kendyala, R., KURMA, J., & Mamidala, J. V. (2021). Enhancing Cloud Infrastructure Security Through AI-Powered Big Data Anomaly Detection. *Available at SSRN 5741305*.

45. Singh, A. A. S., Tamilmani, V., Maniar, V., Kothamaram, R. R., Rajendran, D., & Namburi, V. D. (2021). Predictive Modeling for Classification of SMS Spam Using NLP and ML Techniques. *International Journal of Artificial Intelligence, Data Science, and Machine Learning*, *2*(4), 60-69.

46. Padur, S. K. R. (2020). AI augmented disaster recovery simulations: From chaos engineering to autonomous resilience orchestration. *International Journal of Scientific Research in Science, Engineering and Technology*, *7*(6), 367-378.

47. Reddy Padur, S. K. (2021). From Scripts to Platforms-as-Code: The Role of Terraform and Ansible in Declarative Infrastructure Rollouts. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, 621-628.

48. Kothamaram, R. R., Rajendran, D., Namburi, V. D., Singh, A. A. S., Tamilmani, V., & Maniar, V. (2021). A Survey of Adoption Challenges and Barriers in Implementing Digital Payroll Management Systems in Across Organizations. *International Journal of Emerging Research in Engineering and Technology*, *2*(2), 64-72.

49. Padur, S. K. R. (2018). Autonomous cloud economics: AI driven right sizing and cost optimization in hybrid infrastructures. *International Journal of Scientific Research in Science and Technology*, *4*(5), 2090-2097.

50. Rajendran, D., Namburi, V. D., Singh, A. A. S., Tamilmani, V., Maniar, V., & Kothamaram, R. R. (2021). Anomaly Identification in IoT-Networks Using Artificial Intelligence-Based Data-Driven Techniques in Cloud Environmen. *International Journal of Emerging Trends in Computer Science and Information Technology*, *2*(2), 83-91.

51. Padur, S. K. R. (2021). Bridging Human, System, and Cloud Integration through RESTful Automation and Governance. *the International Journal of Science, Engineering and Technology*, *9*(6).

52. Attipalli, A., BITKURI, V., KURMA, J., Enokkaren, S., Kendyala, R., & Mamidala, J. V. (2021). A Survey of Artificial Intelligence Methods in Liquidity Risk Management: Challenges and Future Directions. *Available at SSRN 5741342*.

53. Padur, S. K. R. (2021). From Control to Code: Governance Models for Multi-Cloud ERP Modernization. *International Journal of Scientific Research & Engineering Trends*, *7*(3).

54. Routhu, K. K. (2021). Harnessing AI Dashboards in Oracle Cloud HCM: Advancing Predictive Workforce Intelligence and Managerial Agility. *International Journal of Scientific Research & Engineering Trends*, *7*(6).

55. Padur, S. K. R. (2021). Deep learning and process mining for ERP anomaly detection: Toward predictive and self-monitoring enterprise platforms. *Available at SSRN 5605531*.

**Citation:** Srikanth Reddy Keshireddy, Venkata Teja Nagumotu, et al., "An AI-Based Framework for Detecting IoT Botnets Through Network Traffic Analysis and Modeling", Universal Library of Engineering Technology, 2023; 42-52. DOI: https://doi.org/10.70315/uloap.ulete.2023.007.