Advanced Machine Learning for Robust Botnet Attack Detection in Evolving Threat Landscapes

*Abstract*—As technology advances and security issues and cyberattacks increase, more and more Internet of Things (IoT) devices are linked to networks, like botnets have been emerging and evolving very fast and they have a very high possible dangerous attack. IoT transition is disrupted using these attacks disrupting the IoT devices' networks and services approaches for botnet attack detection and classification using Machine Learning (ML) and Deep Learning (DL) have been developed within the framework of the IoT. In this study, provide an intrusion detection system (IDS) based on the Bidirectional Gated Recurrent Unit (Bi-GRU) for detecting botnet attacks in IoT networks. We use the N-BaIoT dataset for this purpose. We opted for a Bi-GRU model, which can detect contextual dependencies in the past and the future, to deal with the sequential IoT traffic data. The model's ability to recognize various botnet attack types even in cases of data imbalance was demonstrated by important performance metrics such as ROC-AUC, accuracy, precision, recall, and F1-score. The results show that the proposed Bi-GRU based IDS is a robust and improved solution for detecting IoT botnet attacks on a real time basis.

Keywords—Cybersecurity, Botnet Attacks, Intrusion Detection Systems (IDS), Network Traffic, Bi-GRU, IoT, N-BaIoT Dataset.

# Introduction

Digital infrastructure, driven by innovations in cloud computing, the IoT and high-speed communication networks has rapidly expanded to alter how we govern, work, and live in a world that is becoming more digitally linked. The advancements in these technologies provide unprecedented efficiency and connectivity, but their very advancements have also resulted in an expanded and complex cyber threat landscape [1]. As systems become more complex, scale and, therefore, more complex, cyberattacks mounted against their vulnerabilities also increase. Cyberattacks conducted by botnets networks of hijacked, internet connected devices remotely controlled by malicious actors, commonly called botmasters are among the most persistent and dangerous. Once infected with malware, these devices can be used to perform coordinated attacks from DDoS and phishing to data theft and further propagation of other malware. What makes botnets so insidious is their ability to operate stealthily, self-update, mutate to evade detection and hide behind encrypted or decentralized communications channels [2].

The threat landscape is continually shifting which is truly challenging for cybersecurity practitioners. However, traditional defence mechanisms, including signature based orrulebased IDS, typically are not capable of detecting emergent or obfuscated botnet behaviour in real time [3]. However, their dependency on known patterns and pre-defined rules hampers their ability to identify new threats that botnet attacks have become more sophisticated and adaptive. Consequently, botnet attack detection in the context of dynamic threat landscapes is now an important area of research and practice [4]. Today, more than ever, organization faces the need to detect accurately, saleably and in real time such traffic when their network traffic is growing to massive volumes of changing and heterogeneous content. This means that modern detection now requires systems that can spot such subtle anomalies and contextual deviations that often represent an indicator of malicious activity.

In order to tackle these challenges, ML and DL techniques are getting more and more popular in research. These intelligent systems are able to decipher massive volumes of network data, learn sophisticated patterns and generalize to threats that have not been seen before [5]. Attack detection problems have started using machine learning in them and ML is getting used in more and more fields connected to cybersecurity. It has been shown that ML algorithms have a good performance in detecting botnet behaviour from statistical and behavioural features. In the meantime, DL models provide richer Network traffic spatial and temporal patterns to improve detection precision and reduce FPR.

## Motivation and Contribution of the Study

This study is motivated by the rising ubiquity of IoT device which are notorious for their susceptibility to Miraa and Gadget botnet attacks and the concerns they present. Such attacks pose a challenge for traditional IDS to effectively detect these attacks, because network traffic is often dynamic and diverse. In this research work, an advanced IDS based on Bi-GRU model is proposed to capture past and future contextual information from a dataset of IoT network traffic to detect botnet attacks with higher accuracy which also rectifies the problem of imbalance in the dataset and high dimensional feature set. This research has three key contributions:

* The N-BaIoT dataset was employed with 828,000 records of benign and malicious IoT traffic for model training.
* Effective handling of the imbalanced dataset was achieved by applying pre-processing techniques such as normalization, one hot encoding and feature selection.
* RF Regressor has been used for feature Selection, reducing dimensionality and overfitting to maintain important features for detection.
* Borrowed from the botnet domain, a Bi-GRU model was introduced that captures past and future contextual information to identify botnet attacks on IoT.
* The model's performance for dependable attack detection was assessed using criteria such ROC AUC, F1 score, recall, accuracy, and precision.

## Justification and Novelty of the Paper

This study is interesting and justified since it uses the N-BaIoT dataset to identify botnet assaults on IoT networks using an innovative Bi-GRU, DL model. The Bi-GRU model is excellent at capturing both short-term and long-term temporal relationships in network traffic, in contrast to traditional approaches that frequently find it difficult to manage the dynamic and changing nature of botnet assaults. Through the processing of both forward and backward data, the Bi-GRU enhances the ability to detect complex attack patterns that may be missed by traditional models like MLP and RF, which do not leverage temporal data to the same extent.

## Structure of the paper

The paper is structured as follows: Section II examines relevant research on enhancing the security of IoT networks and detecting botnets, Section III describes the preparation procedures, the dataset, and the suggested Bi-GRU model, Section IV provides performance analysis and experimental findings, and Section V provides the primary findings and suggestions for more research from the study.

# Literature Review

This section reviews and emphasizes botnet attack detection using advanced ML techniques within the context of evolving cybersecurity threats. The comparative analysis of background study based on their Author, Methodology, Data, Key Findings, Limitations and Future work are provided in Table I.

Liu, Liu and Zhang (2019) propose an approach using DL for the detection of botnets in the IoT. Basic traffic aspects of IoT devices are extracted using damped incremental statistics, and the features are normalized using the Z-Score approach. The MCA, TAM is then used to create the dataset. After learning the dataset, they create a CNN and use the CNN to detect traffic. The final test results demonstrate that their method has a 99% accuracy rate in separating benign traffic from various types of attack traffic [6].

Esmaeili and Shahriari (2019) A technique using the PODBot tool has been presented. Both network traffic analysis and application characteristics are used in the detecting process. PODBot showed an accuracy rate of Tested on many well-known botnet variations, the detection rate was about 87% being at high danger, with 96% being at extremely high risk. Additionally, because of the combination of the detection approaches, it has qualities that make it better than the earlier methods in qualitative assessments of similar jobs. The increasing use of smartphones with several sensors makes them a prime target for criminal activity and bot assaults [7].

Ghafir et al. (2018) offers a brand-new method for detecting botnet C&C communication, known as Bidet, to protect vital ultrastructure systems against malware assaults. The suggested system's development is being done in two stages In order to identify potential botnet C&C communication methods, they have created four detection modules. Additionally, a correlation architecture has been created to lessen the number of false alarms generated by distinct detecting units. According to evaluation results, BotDet effectively balances the TPR and FPR, achieving 82.3% and 13.6%, respectively. Additionally, it demonstrates the real-time detection capabilities of BotDet [8].

Nguyen, Ngo and Le (2018) combine an updated approach to identify Linux IoT botnets using a convolutional neural network (CNN) classifier in conjunction with a principal component analysis (PSI) graph. The experiment utilized 10033 ELF files, out of which 6031 were benign files and 4002 were samples from IoT botnets. Testing shows that the PSI graph CNN classifier achieves an F-measure of 94% and an accuracy rate of 92%. IoT devices are being used more and more for a variety of purposes and in a wider range of domains. They are becoming more prevalent in a variety of applications because to their processing and computational power, which has made them a desirable target for assaults using IoT botnet malware. ML has shown to be a useful technique for malware detection specialists in recent years [9].

Pradeepthi and Kannan (2018) The proposal is for using neural fuzzy classification algorithms as a novel approach to identify botnet traffic. A dataset was generated by constructing an application on Eucalyptus Cloud and then assaulting it using many open-source botnet simulation tools in order to evaluate the strategy. The accuracy of the algorithm was 94.78% using 15,000 cases and 56 attributes. The system's FP are much fewer than those of other similar systems since fuzzy rules have been included into it [10].

Al-Nawasrah et al. (2018) a novel system, the FFKS, which uses an implementation built on an ADeSNN to identify FF-Domains in online mode. When compared to existing algorithms, the suggested approach demonstrated a high detection accuracy (98.77%) in online mode for FF domains, with low rates of false positives and negatives, respectively. A high degree of performance has also been demonstrated. Furthermore, the suggested algorithmic modification improved and aided in the process of customizing the parameters [11]

1. Table 1: Comparative Analysis of Literature Studies on Machine Learning-Based Botnet Attack Detection

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Author(s)** | **Methodology** | **Data** | **Key Findings** | **Limitations and Future Work** |
| Liu, Liu, and Zhang (2019) | Deep learning approach using damped incremental statistics for feature extraction and CNN for detection | IoT traffic data with damped incremental statistics, Z-Score normalization | Achieved 99% accuracy in distinguishing benign traffic and various botnet traffic types | Limited to specific IoT traffic and attacks. Future work: Extend to more IoT environments and refine CNN model |
| Esmaeili and Shahriari (2019) | PODBot tool, which combines network traffic analysis and application functionalities | Botnet data with application features and network traffic | detected 96% of extremely high-risk botnets and over 87% of high-risk botnets. | Limited to mobile phone attacks. Future work: Expand detection scope to include new botnet types and environments |
| Ghafir et al. (2018) | BotDet system, which includes modules for detection and correlation framework for C&C traffic | Traffic information for command and control (C&C) | Real-time detection with a balanced proportion of false positives (13.6%) and true positives (82.3%) | High false positive rate in certain scenarios. Future work: Improve false alarm reduction and scalability in real-time systems |
| Nguyen, Ngo, and Le (2018) | CNN classifier for PSI graph-based Linux IoT botnet detection | In terms of a total of 10033 ELF files, 6031 are considered innocuous, whereas 4002 are examples of IoT botnets | Achieved 92% accuracy and 94% F-measure in IoT botnet detection | Focused on Linux-based IoT devices. Future work: Extend to other IoT platforms and enhance PSI graph method |
| Pradeepthi and Kannan (2018) | Neuro-fuzzy classification techniques for botnet traffic detection | 15,000 instances with 56 attributes from Eucalyptus cloud | Achieved 94.78% accuracy and reduced false positives with fuzzy rules | Limited to cloud-based botnet attacks. Future work: Explore hybrid models and other datasets for improved detection |
| Al-Nawasrah et al. (2018) | ADeSNN is a dynamically evolving spike neural network that is adaptable, is used for the identification of fast flux domains (FF-Domains). | FF-Domains in online mode | Achieved 98.77% detection accuracy with low false positives and negatives | Focused only on FF-Domains. Future work: Apply the system to other attack types and improve real-time adaptation |

# Methodology

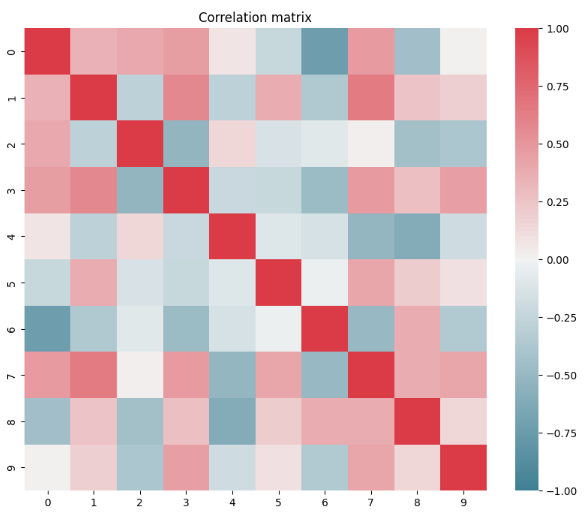
The methodology for ML for Robust Botnet Attack Detection involves a structured approach shown in Figure 1. The methodology begins with gathering information about network traffic from the N-BaIoT dataset. Data pre-processing is performed to enhance quality through the removal of null and duplicate entries, categorical feature one-hot encoding and Z-score normalization for input that is standardized. The RF Regressor is used to select features in order to reduce dimensionality, and the data 80% goes towards training and 20% towards testing. Crucial actions ROC-AUC, F1-score, recall, accuracy, and precision are some of the measures used to assess the Bi-GRU model, which processes sequences both forward and backward to capture complicated spatiotemporal connection, ensuring robust performance in detecting attacks amidst the imbalanced nature of the dataset.

1. Flowchart for Botnet Attack Detection

The overall steps of the flowchart for Botnet Attack Detection are provided below:

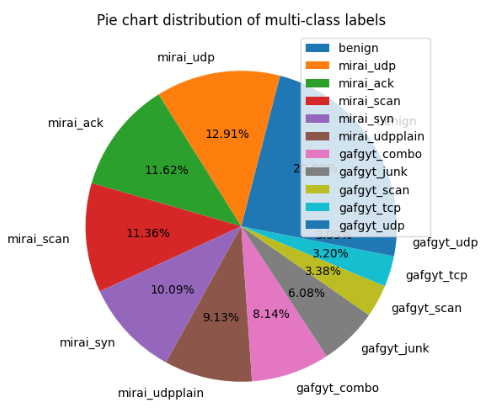
## Data Collection

In this investigation, the N-BaIoT dataset is utilized, with a focus on network traffic data from the Provision PT-737E IoT security camera. The dataset comprises over 828,000 records, including both benign traffic and malicious activity generated by Miraa and Gadget botnet families, covering 10 distinct attack types. Each data instance includes 115 statistical features extracted across 5-time windows, providing a rich foundation for spatiotemporal analysis and accurate intrusion detection. Some visualization of data given are:



1. Correlation Matrix

The correlation matrix heatmap in Figure 2 illustrates the connections between 10 factors. Strong negative correlations are shown by blue, weak or no correlation is shown by white, and strong positive correlations are shown by red. For complete self-correlation, the diagonal values are 1. Multicollinearity or relationships between variables in the dataset may be found with the use of this visual aid.



1. Pie Chart for Distribution of Labels

Figure 3 shows the label distribution in the dataset using a pie chart. 'Benign' traffic makes up the largest portion at 25.08%, followed by major such as "mirai\_ack," "mirai\_udp," and "mirai\_scan" attacks. The other groups, such as "gafgyt\_junk," "gafgyt\_scan," "gafgyt\_tcp," and "gafgyt\_udp," include smaller but still significant portions of the data. This provides an imbalanced distribution that will be an important consideration when creating and evaluating intrusion detection models.

## Data Preprocessing

Data pre-processing is a cleansing procedure that transforms raw, unstructured data into clean, well-structured data for additional study. Removing null and duplicate values, one hot encoding for categorical features and Z score normalization to standardize the data are all data pre-processing. Taking these steps enables us to have clean, consistent and scale data which in turn help our model perform better and be more stable. Below I provide the key steps of pre-processing:

* **Removal of Null or Missing Values:** To avoid training issues, any entries with missing (NaN) values were removed.
* **Elimination of Duplicate Records:** To prevent bias in our training dataset we identified and dropped duplicate rows in the dataset.

## Data Encoding with One-Hot Encoding

To combat power consumption due to transitions in the chip's connections, we design data encoding techniques.   
One-hot encoding is commonly utilized in jobs requiring numerical input from ML methods like neural networks and LR. The formula is shown in Equations (1) or (2).

(1)

Where,

(2)

The encoded binary value represents category j, where i is the index of the categorical value among n unique categories. The encoding assigns a binary vector of length n, setting [12] the position corresponding to the category to 1, while all other positions remain 0.

## Data Normalization with Z-Score Standardization

Normalization is a scaling technique that converts a large feature set into a regular range. It normally falls between 0 and 1. To create a data set, set the standard deviation to one and the mean to zero. Using this scaling technique is beneficial when the data has a normal distribution; otherwise, it will cause complications. The standardized value X′ is calculated using Equation (3).

(3)

where the dataset mean is represented by μ, the standard deviation by σ, and the original value by X [13].

## Feature selection using RF

To enhance model performance, Feature selection is the process of discovering and selecting the most relevant attributes from a dataset. It reduces training time, avoids overfitting, and simplifies the issue. To choose features, utilize the RF Regressor function. Tests have shown that RF regressors are an excellent way to decrease a dataset's number of variables.

## Data Splitting

Data Splitting, you take a dataset, split it into different subsets and use these subsets to train and test your data. 20% of the Bot-IoT data was utilized for the purpose of testing the model's performance, with the remaining 80% being utilized for training.

## Propose Bidirectional Gated Recurrent Unit (Bi-GRU) Model

The Bi-GRU version of the GRU network is created to handle sequences that go backwards as well as forwards. While traditional GRUs look at only previous information, Bi-GRU takes into account context from both past and future information [14][15]. Two separate GRU networks are employed: one reads the data forward from the first to the last point in time and the other reads backward, in reverse order. For every time step, both networks’ outputs are merging to create the final result. Mathematically, the forward GRU updates the hidden state ​​ as: given in Equation (4).

(4)

and the backward GRU updates the hidden state ​​ as: shown given Equation (5).

(5)

The hidden states for forward and backward motion are combined to obtain the last concealed condition for each time step: This is computed using Equation (6).

(6)

Their bidirectional approach can capture more complex patterns in sequential data and takes better advantage of past and future context [16]. In applications such as time series forecasting, sentiment analysis and botnet attack detection, bi GRU has been extensively used due to its ability to archive better performance with bidirectional information of elements preceding and succeeding the sequence.

## Performance Matrix

The purpose of the performance matrix is to evaluate an ML model's effectiveness using important performance indicators such as TP, TN, FP, and FN, which are derived from the misinterpretation matrix. Nonetheless, these metrics assess the capacity of the model to distinguish between valid traffic and malicious ones [17]. Recall is the reciprocity of identifying real assaults, accuracy measures overall correctness, precision rate is the ratio of properly detected attacks, and F1 Score is calculated by averaging recall and accuracy. The fundamental parameters are:

* **True Positive (TP):** This attack's corresponding class characteristic has been appropriately detected by the classifier.
* **True Negative (TN):** The negative value of the class characteristic indicates that there is normal traffic.
* **False Positive (FP):** The classifier incorrectly identifies normal traffic as malicious.
* **False Negative (FN):** The classifier makes the incorrect assumption that an attack record is simply normal traffic.

### Accuracy

The most well-known performance metric is accuracy. It is simple and convenient to compute and identify. The formula is shown in Equation (7).

(7)

### Precision

The precision shows how accurate the classification is. Both high precision and poor accuracy lead to fewer FP and lower accuracy. It is determined by using Equation (8):



### Recall

Sensitivity can be described by the TPR or recall. There are a few easy ways to rapidly determine the TP percentage. It is given blow Equation (9).

(9)

### F1-Score

F1-Measure combines precision with sensitivity this is the weighted harmonic method for accuracy and precision and sensitivity. It has been demonstrated that the F1 measurement is just as accurate as. It is provided below Equation (10).



### Roc-curve

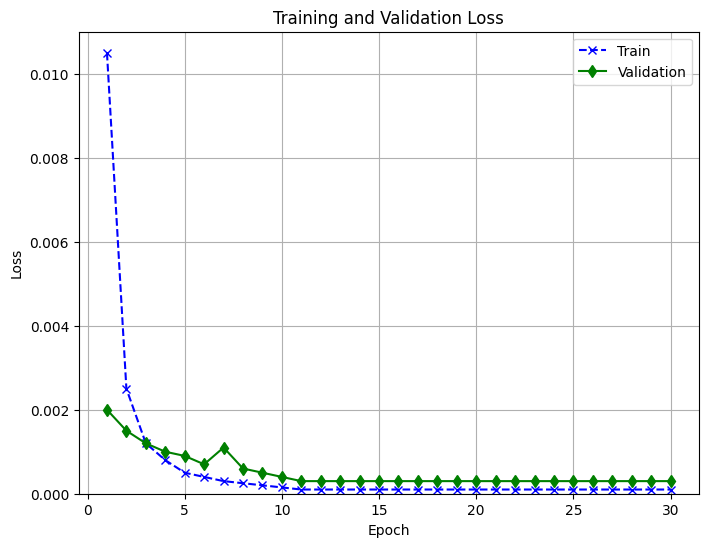
The integral of the FPR times the TPR yields the area under the ROC and AUC. AUC is a metric value that continuously falls between 0 and 1.

# Result Analysis And Discussion

In this section, the results of the suggested Bi-GRU model used to identify botnet assaults in IoT networks. Python 3 with TensorFlow and Scikit-learn were used for the experiments using a machine running 64-bit Windows 10 with 16 GB of random-access memory and an Intel i7 CPU (3.60 GHz, four cores). The Bi-GRU model performance, shown in Table II, achieved exceptional results in classifying network traffic. The system's accuracy in identifying both malicious and benign traffic was 99.99%. Our 99.98% recall and accuracy ensure that there aren't many FP and that the system can really identify threats. Another indication of robustness and dependability is an F1 score of 99.93%, which achieves equilibrium between accuracy and recall. The outcomes show that the model is dependable for deployment in extremely dynamic IoT contexts since it can accurately detect botnet assaults.

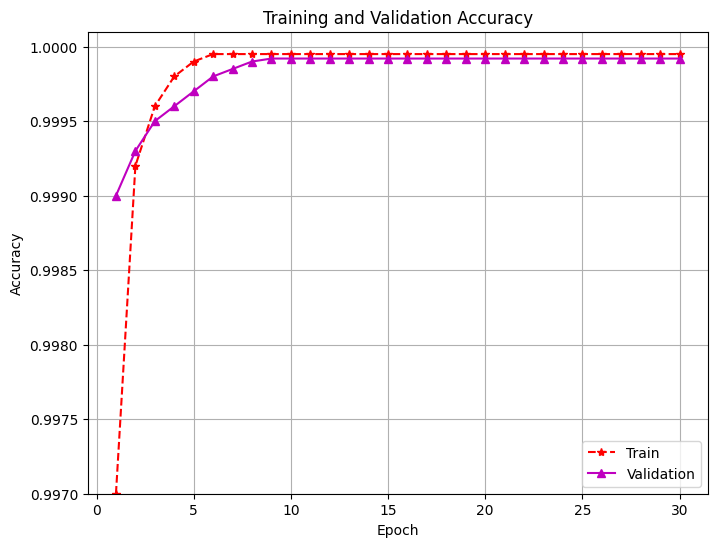
1. Experiment Result of the Bi-GRU Model for Botnet Attack Detection

|  |  |
| --- | --- |
| **Evaluation Matrix** | **Bidirectional Gated Recurrent Unit (Bi-GRU)** |
| Accuracy | 99.99 |
| Precision | 99.98 |
| Recall | 99.98 |
| F1-Score | 99.93 |



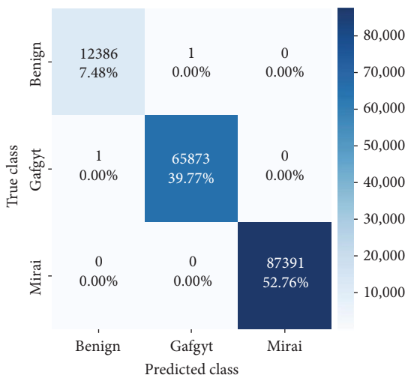
1. Plot loss Curve of the Bi-GRU

The loss of the Bi-GRU model over 30 epochs in training and validation are illustrated in Figure 4. Both losses eventually converge near zero very quickly. Therefore, the good generalization and the absence of overfitting is confirmed by the minimal gap between curves and shows that the model is able to detect anomalies.



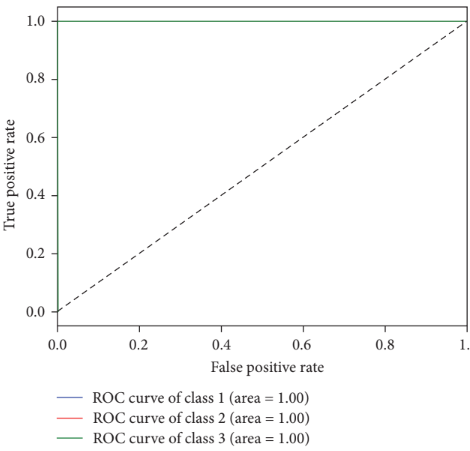
1. Plot Accuracy Curve of the Bi-GRU

Figure 5 shows the accuracy of the Bi-GRU model training and validation across 30 epochs. Additionally, the accuracy of these models rapidly rises and eventually levels out at almost 100%, indicating strong model performance. Plots of the validation and training curves are quite near to one another. suggesting a strong capability for generalization and absence of overfitting to the classes during the learning.



1. Confusion Matrix for the Bi-GRU

Figure 6 displays the Bi-GRU model's confusion matrix, showing excellent classification performance across three classes: Benign, Gafgyt, and Mirai. The model correctly classifies nearly all samples with minimal misclassifications, achieving particularly high accuracy in detecting Mirai (52.76%) and Gafgyt (39.77%) attacks.



1. ROC curves for the Bi-GRU

Figure 7 illustrates the ROC curves for the Bi-GRU model across all three classes (Benign, Gafgyt, Mirai), each achieving a perfect AUC of 1.00. This indicates outstanding model performance with perfect discrimination capability, achieving a 100% TPR and a 0% FPR.

## Comparative Analysis

A comparison of various this section offers machine learning algorithms for identifying botnet assaults in Internet of Things networks. Table III compares the accuracy of many methods used to categories assaults on the dataset for N-BaIoT. The MLP model's accuracy rate of 84% was surpassed by 96% by the RF model. Nevertheless, the Bi-GRU model outperformed all others with the accuracy rate of 99.9% which proves its effectiveness in identifying long range dependence in network traffic and accurately detecting botnet attacks.

1. Comparative Analysis between Models Performance for Botnet Attack Detection

|  |  |
| --- | --- |
| **Models** | **Accuracy** |
| Multilayer Perceptron (MLP)[18] | 84 |
| Random Forest (RF)[19] | 96 |
| Proposed Bi-GRU | 99.99 |

The proposed methodology presents several advantages, among which the application of the Bi-GRU model allows to precisely simulate the complex spatiotemporal patterns of network traffic and thus to provide very accurate botnet attack detection. The system pre-processes input robustly, uses RF Regressor for effective feature selection and normalizes the inputs standardly, thereby achieving high data quality and model efficiency. It achieved 99.9% accuracy and better than traditional model like MLP, RF and Bi-LSTM. It also has the capacity to deal with imbalanced data and deliver early threat detection, securing and making IoT networks reliable at same time.

# Conclusion And Future Scope

Botnet attacks in modern IoT networks can cause serious disruption and security breach which in turns brings serious risk and large operational cost. ML powered botnet attack detection is an effective solution to predict and identify malicious activities before they can get out of hands. Only by early detection can IoT systems be kept secure and intact. Performance criteria like as Training and verifying the Bi-GRU model has been done using F1 score, recall, accuracy, and precision, which has shown an amazing 99.9% accuracy rate in differentiating between malicious and benign traffic. The Bi-GRU model is compared with other traditional ML models, (MLP, RF, Bi-LSTM) and shows its strong capability in terms of modelling complex attack patterns. While the model performs impressively, it has some problems, including the minor misclassification in complex attack cases and dependency on a single dataset which restricts its generalization. Future work will focus on improving model robustness by exploring domain adaptation techniques, ensemble learning, and hybrid architectures to better handle evolving threats in real-world IoT environments.

##### References

[1] K.-K. R. Choo, “The cyber threat landscape: Challenges and future research directions,” *Comput. Secur.*, vol. 30, no. 8, pp. 719–731, 2011, doi: https://doi.org/10.1016/j.cose.2011.08.004.

[2] S. Singamsetty, “Fuzzy-Optimized Lightweight Cyber-Attack Detection For Secure Edge-Based Iot Netwok.,” *J. Crit. Rev.*, vol. 6, no. 7, 2019, doi: 10.53555/jcr.v6:i7.13156.

[3] K. Sinha, A. Viswanathan, and J. Bunn, “Tracking Temporal Evolution of Network Activity for Botnet Detection,” 2019.

[4] S. Anwar, J. M. Zain, M. F. Zolkipli, and Z. Inayat, “A Review Paper on Botnet and Botnet Detection Techniques in Cloud Computing,” 2014.

[5] S. Haq and Y. Singh, “Botnet detection using machine learning,” *PDGC 2018 - 2018 5th Int. Conf. Parallel, Distrib. Grid Comput.*, pp. 240–245, 2018, doi: 10.1109/PDGC.2018.8745912.

[6] J. Liu, S. Liu, and S. Zhang, “Detection of IoT botnet based on deep learning,” in *Chinese Control Conference, CCC*, 2019. doi: 10.23919/ChiCC.2019.8866088.

[7] S. Esmaeili and H. R. Shahriari, “PodBot: A New Botnet Detection Method by Host and Network-Based Analysis,” in *ICEE 2019 - 27th Iranian Conference on Electrical Engineering*, 2019. doi: 10.1109/IranianCEE.2019.8786432.

[8] I. Ghafir *et al.*, “BotDet: A System for Real Time Botnet Command and Control Traffic Detection,” *IEEE Access*, vol. 6, pp. 38947–38958, 2018, doi: 10.1109/ACCESS.2018.2846740.

[9] 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.

[10] K. V Pradeepthi and A. Kannan, “Detection of Botnet traffic by using Neuro-fuzzy based Intrusion Detection,” *2018 Tenth Int. Conf. Adv. Comput.*, pp. 118–123, Dec. 2018, doi: 10.1109/ICoAC44903.2018.8939109.

[11] A. Al-Nawasrah, A. Al-Momani, F. Meziane, and M. Alauthman, “Fast flux botnet detection framework using adaptive dynamic evolving spiking neural network algorithm,” in *2018 9th International Conference on Information and Communication Systems (ICICS)*, 2018, pp. 7–11. doi: 10.1109/IACS.2018.8355433.

[12] J. Li, Y. Si, T. Xu, and S. Jiang, “Deep Convolutional Neural Network Based ECG Classification System Using Information Fusion and One-Hot Encoding Techniques,” *Math. Probl. Eng.*, vol. 2018, no. 1, pp. 1–10, Dec. 2018, doi: 10.1155/2018/7354081.

[13] J. M. A. Peshawa and H. F. Rezhna, “Data Normalization and Standardization: A Technical Report​,” *Mach. Learn. Tech. Reports*, vol. 1, no. 1, pp. 1–6, 2014, doi: 10.13140/RG.2.2.28948.04489.

[14] R. Yi and W. Hu, “Pre-Trained BERT-GRU Model for Relation Extraction,” in *Proceedings of the 2019 8th International Conference on Computing and Pattern Recognition*, Oct. 2019, pp. 453–457. doi: 10.1145/3373509.3373533.

[15] R. Tarafdar and Y. Han, “Finding Majority for Integer Elements,” *J. Comput. Sci. Coll.*, vol. 33, no. 5, pp. 187–191, 2018.

[16] T. Yongwang, L. Xin, and D. Qizheng, “Malicious Code Detection Technology based on Bi-GRU and Self-attention,” in *Proceedings of the 2019 2nd International Conference on Algorithms, Computing and Artificial Intelligence*, Dec. 2019, pp. 585–590. doi: 10.1145/3377713.3377793.

[17] D. P. de O. Godeiro, M. L. R. Dantas, M. dos S. Celestino, and D. C. Da Silva, “Application of Importance and Performance Matrix to Assess the Quality of Services Provided by Business Incubators,” *REGEPE Entrep. Small Bus. J.*, vol. 7, no. 3, pp. 01–29, Sep. 2018, doi: 10.14211/regepe.v7i3.704.

[18] J. Alsamiri and K. Alsubhi2, “Internet of things cyber attacks detection using machine learning,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 10, no. 12, p. 627, 2019, doi: 10.14569/ijacsa.2019.0101280.

[19] 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.

Kalla, D., Smith, N., & Samaah, F. (2023). Satellite Image Processing Using Azure Databricks and Residual Neural Network. *International Journal of Advanced Trends in Computer Applications*, *9*(2), 48-55.

Kuraku, D. S., & Kalla, D. (2023). Phishing Website URL’s Detection Using NLP and Machine Learning Techniques. *Journal on Artificial Intelligence-Tech Science*.

Varadharajan, V., Smith, N., Kalla, D., Samaah, F., Polimetla, K., & Kumar, G. R. (2024). Stock Closing Price and Trend Prediction with LSTM-RNN. *Journal of Artificial Intelligence and Big Data*, *4*, 877.

Kalla, D., Kuraku, D. S., & Samaah, F. (2021). Enhancing cyber security by predicting malwares using supervised machine learning models. *International Journal of Computing and Artificial Intelligence*, *2*(2), 55-62.

Kuraku, D. S., Kalla, D., Smith, N., & Samaah, F. (2023). Safeguarding FinTech: elevating employee cybersecurity awareness in financial sector. *International Journal of Applied Information Systems (IJAIS)*, *12*(42).

Kalla, D., Smith, N., Samaah, F., & Polimetla, K. (2022). Enhancing Early Diagnosis: Machine Learning Applications in Diabetes Prediction. *Journal of Artificial Intelligence & Cloud Computing. SRC/JAICC-205. DOI: doi. org/10.47363/JAICC/2022 (1)*, *191*, 2-7.

Kuraku, S., Kalla, D., Samaah, F., & Smith, N. (2023). Cultivating proactive cybersecurity culture among IT professional to combat evolving threats. *International Journal of Electrical, Electronics and Computers*, *8*(6).

Kalla, D., Smith, N., Samaah, F., & Polimetla, K. (2024). Hybrid Scalable Researcher Recommendation System Using Azure Data Lake Analytics. *Journal of Data Analysis and Information Processing*, *12*, 76-88.

Kuraku, D. S., & Kalla, D. (2023). Impact of phishing on users with different online browsing hours and spending habits. *International Journal of Advanced Research in Computer and Communication Engineering*, *12*(10).

Kalla, D., & Kuraku, S. (2023). Phishing website url’s detection using nlp and machine learning techniques. *Journal of Artificial Intelligence*, *5*, 145.

Kuraku, D. S., Kalla, D., & Samaah, F. (2022). Navigating the link between internet user attitudes and cybersecurity awareness in the era of phishing challenges. *International Advanced Research Journal in Science, Engineering and Technology*, *9*(12).

Kuraku, D. S., Kalla, D., Smith, N., & Samaah, F. (2023). Exploring How User Behavior Shapes Cybersecurity Awareness in the Face of Phishing Attacks. *International Journal of Computer Trends and Technology*.

Sreeramulu, M. D., Mohammed, A. S., Kalla, D., Boddapati, N., & Natarajan, Y. (2024, September). AI-driven Dynamic Workload Balancing for Real-time Applications on Cloud Infrastructure. In *2024 7th International Conference on Contemporary Computing and Informatics (IC3I)* (Vol. 7, pp. 1660-1665). IEEE.

Kalla, D., Mohammed, A. S., Boddapati, V. N., Jiwani, N., & Kiruthiga, T. (2024, November). Investigating the Impact of Heuristic Algorithms on Cyberthreat Detection. In *2024 2nd International Conference on Advances in Computation, Communication and Information Technology (ICAICCIT)* (Vol. 1, pp. 450-455). IEEE.

Chandrasekaran, A., & Kalla, D. (2023). Heart disease prediction using chi-square test and linear regression. *Computer Science & Information Technology*, *13*, 135-146.

Chinta, P. C. R., Katnapally, N., Ja, K., Bodepudi, V., Babu, S., & Boppana, M. S. (2022). Exploring the role of neural networks in big data-driven ERP systems for proactive cybersecurity management. *Kurdish Studies*.

Routhu, K., Bodepudi, V., Jha, K. M., & Chinta, P. C. R. (2020). A Deep Learning Architectures for Enhancing Cyber Security Protocols in Big Data Integrated ERP Systems. *Available at SSRN 5102662*.

Moore, C. (2023). AI-powered big data and ERP systems for autonomous detection of cybersecurity vulnerabilities. *Nanotechnology Perceptions*, *19*, 46-64.

Bodepudi, V., & Chinta, P. C. R. (2024). Enhancing Financial Predictions Based on Bitcoin Prices using Big Data and Deep Learning Approach. *Available at SSRN 5112132*.

Chinta, P. C. R. (2023). The Art of Business Analysis in Information Management Projects: Best Practices and Insights. *DOI*, *10*.

Boppana, S. B., Moore, C. S., Bodepudi, V., Jha, K. M., Maka, S. R., & Sadaram, G. AI And ML Applications In Big Data Analytics: Transforming ERP Security Models For Modern Enterprises.

Katnapally, N., Chinta, P. C. R., Routhu, K. K., Velaga, V., Bodepudi, V., & Karaka, L. M. (2021). Leveraging Big Data Analytics and Machine Learning Techniques for Sentiment Analysis of Amazon Product Reviews in Business Insights. *American Journal of Computing and Engineering*, *4*(2), 35-51.

Chinta, P. C. R., Moore, C. S., Karaka, L. M., Sakuru, M., Bodepudi, V., & Maka, S. R. (2025). Building an Intelligent Phishing Email Detection System Using Machine Learning and Feature Engineering. *European Journal of Applied Science, Engineering and Technology*, *3*(2), 41-54.

Moore, C. (2024). Enhancing Network Security With Artificial Intelligence Based Traffic Anomaly Detection In Big Data Systems. Available at SSRN 5103209.

Chinta, P. C. R., Moore, C. S., Karaka, L. M., Sakuru, M., & Bodepudi, V. (2025). Predictive Analytics for Disease Diagnosis: A Study on Healthcare Data with Machine Learning Algorithms and Big Data. *J Cancer Sci*, *10*(1), 1.

KishanKumar Routhu, A. D. P. Risk Management in Enterprise Merger and Acquisition (M&A): A Review of Approaches and Best Practices.

Bodepudi, V. (2023). Understanding the Fundamentals of Digital Transformation in Financial Services: Drivers and Strategic Insights. *Journal of Artificial Intelligence and Big Data*, *3*(1), 10-31586.

Chinta, P. C. R. (2022). Enhancing Supply Chain Efficiency and Performance Through ERP Optimisation Strategies. *Journal of Artificial Intelligence & Cloud Computing*, *1*(4), 10-47363.

Krishna Madhav, J., Varun, B., Niharika, K., Srinivasa Rao, M., & Laxmana Murthy, K. (2023). Optimising Sales Forecasts in ERP Systems Using Machine Learning and Predictive Analytics. *J Contemp Edu Theo Artific Intel: JCETAI-104*.

Jha, K. M., Velaga, V., Routhu, K., Sadaram, G., Boppana, S. B., & Katnapally, N. (2025). Transforming Supply Chain Performance Based on Electronic Data Interchange (EDI) Integration: A Detailed Analysis. *European Journal of Applied Science, Engineering and Technology*, *3*(2), 25-40.

Sadaram, G., Sakuru, M., Karaka, L. M., Reddy, M. S., Bodepudi, V., Boppana, S. B., & Maka, S. R. (2022). Internet of Things (IoT) Cybersecurity Enhancement through Artificial Intelligence: A Study on Intrusion Detection Systems. *Universal Library of Engineering Technology*, (2022).

Maka, S. R. (2023). Understanding the Fundamentals of Digital Transformation in Financial Services: Drivers and Strategic Insights. *Available at SSRN 5116707*.