Intrusion Detection System – IIDS

AMJAD JUMAAH FRHAN
Department of Education and Islamic Studies, Sunni Endowment Diwan,
University of AL Mashreq, Department of Cybersecurity Engineering Technology
Baghdad,
IRAQ

Abstract: - The swift evolution of the Internet of Things (IoT) is generating significant cybersecurity challenges, as securing interlinked devices is of the utmost importance. The suggested smart system offers a method that could contribute to improved detection of cyberattacks aimed at IoT devices. This system leverages Conditional Generative Adversarial Networks (CGANs) to produce synthetic attack data, and then the LightGBM algorithm extracts patterns of malicious activity. The smart system accounts for a data preparation stage for IoT devices, and LightGBM's important feature selection step optimizes performance. The smart system is capable of differentiating between multiple attack types that include denial of service (DoS) attacks, address resolution protocol (ARP) spoofing, and data leakage. Using a gradient boosting framework, the smart system provides a reasonable trade-off between computational efficiency and accurate detection performance; furthermore, the model improves upon existing intrusion detection systems using the RT-IoT2022 dataset, attaining 87% accuracy while detecting attacks against the RT-IoT2022 dataset that contained synthetic data created by a GAN. This research demonstrates the capability of the tree-guided gradient boosting method to have effective performance and sound application in resource-constrained environments such as the IoT. It enhances detection capabilities and reduces computational costs. The model exhibited robust effectiveness for intrusion detection, in a time where our digital world has become more connected and more vulnerable, suggesting a more scalable approach was examined with an 80-20 training-test split toward improving cybersecurity for enterprise. The research will further continue to adapt to evolving malware attack strategies and to encourage the actual implementation of results on the Internet of Things.

Key-Words: - Intrusion Detection Systems (IDS), Cyber-Security, Machine Learning, IoT, LightGBM Model, Semi-Supervised Learning.

Received: April 7, 2025. Revised: June 29, 2025. Accepted: August 2, 2025. Published: January 9, 2026.

1 Introduction

Due to the rapid growth of the Internet, networked computer systems have become increasingly indispensable within our society. Although the Internet has many benefits, it also features a downside. In particular, people and organizations who attack and abuse computer systems are presenting new threats each day. Compared to typical intrusion detection systems (IDS), signature-based IDS utilize artificial intelligence and data-driven models to identify, classify, and respond to malicious behavior across different networks, increasing detection accuracy and adaptability, [1].

Cybersecurity issues have become a top concern for businesses. Modern network infrastructures rely heavily on innovation, connectivity, and emerging technologies. As a result, cyberattacks often disrupt business operations and lead to substantial financial losses, [2]. On the other hand, advances in machine learning and artificial intelligence have assisted researchers in handling a variety of issues. This paper's goal is to use some of these technologies to address cybersecurity issues. We examine and evaluate transactional data, create models, and generate predictions from them to identify threats, [3].

Connected technologies like 5G communication, smart grids, and the Internet of Things (IoT) have advanced quickly in recent years, [4]. Although there are many advantages of these changes, the communications networks are much more complex, and the number of devices connected has dramatically increased, increasing the breadth of security threats; for example, health data can now be transferred without any vetting or even review of its source, [5].

There are hurdles in deploying machine learning-based intrusion detection systems. The quality and variety of training datasets, feature selection, and model optimization will all affect the

outcome of the systems. The significance of intrusion detection today is compounded by the high degree of efficiency in handling more data, [6]. Machine learning (ML) models offer superior analytics, can solve problems using data, and have detection capabilities for potential breach indicators. ML approaches offer an alternative path for attack detection in that they can become aware of new and changing threats and thus are proactive vs standard techniques. Using historical data, models can be trained to recognize new and existing attack |events|. This improves the system's accuracy for identifying the variety of breaches, [7]. The goal of security threat detection is to identify suspicious or malicious activity that may indicate an ongoing cyberattack or hacking attempt, understand the nature of attacks and the vulnerabilities they exploit, and improve your security posture by implementing better preventative measures. The paper is structured as follows: Intrusion detection systems are discussed in Section 2, and the data set and preprocessing techniques are analyzed in Section 4. The IDS architecture is described in Section 5, and performance findings are presented in Section 6.

The Contribution can be summarized in the following:

1. A hybrid phased inference framework is developed to enable accurate and scalable intrusion detection by combining traffic-level and packet-level studies.
2. A robust and lightweight detection algorithm is incorporated into the framework, reducing false positives and enhancing performance consistency across different IoT devices.
3. This study involves developing different amounts of attack samples and subsequently integrating them into the original dataset in a manner different from previous studies.

2 Related Work

The research in the field of biological studies has been conducted in depth, using various and diverse methods, and the results have varied according to the circumstances. Below are some of the key areas of interest for this study in recent years.

The authors of this paper propose a lightweight framework, S2CGAN-IDS. This framework leveraging the characteristics of network traffic distribution to enhance the volume of minority classes in both the data space and feature space. The objective is increased minority class detection rate with maintained majority class detection fidelity. Using the CICIDS2017 dataset, the proposed approach was shown to be effective in mitigating the effect of scarcity in the experiments. The experimental results demonstrated that the proposed method outperformed the baseline approach in terms of precision and recall, with a difference of 10.2% in the F1 score. However, the prominent disadvantages of this approach were the high resource usage and the fact that intrusion detection systems themselves are inherently unreliable, [8].

In this work, the author proposed a distributed GAN (generative adversarial network) to create a completely distributed intrusion detection system (IDS) for Internet of Things (IoT) devices, which could detect anomalous behavior without any assistance from a centralized controller, which is important as it can mitigate the identification of both internal or external attacks. The simulation results showed that the proposed DID could achieve improved intrusion detection accuracy compared to an individual IDS implementation that used the same dataset of IoT devices. More specifically, the results of the study showed that the GAN-based DID achieved up to a 20% increase in accuracy, 25% increase in precision, and a 60% decrease in false positive rate compared to a standalone GAN-based IDS, [9].

In this study, the two-stage monitoring scheme was designed to protect the Internet of Things (IoT) setting based on two detectors. Using generative adversarial neural networks (GANs), the researchers first proposed a new type of training. This technique was intended to allow the first detector to train on robust characteristics, whereas the detector was provided with competitive instances that constituted the "validation set." The benefits of the predicted training technique were reported in terms of detection accuracy as well as resilience to adversarial influences. The empirical evidence based on a new dataset on cyberattacks supported that the suggested technique had the potential for detection of both intrusions and continuous adversarial attacks on the IoT and the machine learning capabilities associated with the IoT. The weighted precision, recall, F1-score and accuracy were 96%, 95%, 95%, and 95% respectively, [10].

In this study, the researcher introduced an intrusion detection system based on a conditional generative tabular network (CTGAN) to detect Denial-of-Service (DoS), and Distributed Denial-of-Service (DDoS) attacks in Internet of Things (IoT) networks. The intrusion detection system implements a generative network to collect synthetic network traffic data to enable the generated synthetic tabular data to train different shallow and deep learning classifiers and improve the model’s

detection performance. The Bot-IoT dataset was chosen as the case study to evaluate the proposed methodology measuring F1 score, recall, precision and detection accuracy. The findings showed that the proposed methodology is very accurate in the detection of DoS and DDoS attacks in IoT networks, [11].

This study introduces a hybrid anomaly detection system, referred to as a hybrid DoS Generalized Intrusion Detection System, to detect both known and unknown denial of service attacks and botnets. The hybrid system uses both anomaly-based and signature-based detection. One of the most significant contributions from this study, is a new anomaly detection model proposed called CL-GAN, that includes convolutional neural networks, long-short-term memory networks, and generative adversarial networks to develop a baseline model of network behaviour to improve the detection of malicious traffic. The model improves overall accuracy of detection, while reducing the time to train and test the model. Experimental results demonstrated that the hybrid DoS Generalized Intrusion Detection System model performed better than other intrusion detection systems, while measuring an overall 5% improvement over accuracy, precision, recall, and F1 score in decision making when compared to marketplace intrusions detection systems in multiple diverse datasets NSL-KDD, CICIDS2018, and Bot-IoT, [12]. In this study, the researcher proposed a novel intrusion detection system (IDS) known as LightShield to be used in an IoT context leveraging high-performance computing. The LightShield system combines the preprocessing of IoT data and feature selection utilizing the ReliefF algorithm, along with a new detection model based on the LightGBM framework, which is optimized for gradient-based learning. The proposed system utilizes a graphics processing unit (GPU) for the training process and the binary classification model achieved an accuracy of 99.82% in detecting potential attacks, while the multi-class classification model obtained an accuracy of 97.25% in classifying different attack types, [13].

In another study, the researcher proposed a hybrid IDS utilizing the LightGBM algorithm for anomaly filtering with traffic combination and the MobileNetV2 model for packet classification. The proposed hybrid NNIDS model showed superior execution than competing intrusion detection models in the research on the ACI-IoT-2023 dataset with an accuracy of 94%, F1-score of 91%, and a precision rate of 93%. The experimental results demonstrated that the asymmetric algorithm provided a significant processing cost reduction and possible future application in IoT, [14].

In this study, the researcher proposed an intrusion detection system that integrates a denoising autoencoder and a LightGBM classifier. The LightGBM classifier is used to classify samples, and the proposed model demonstrates significantly improved detection performance compared to other existing intrusion detection systems on nine benchmark datasets for binary and multiclassification tasks. The maximum detection rate of the model has been higher than 99.60% for CIDDS-001, 99.90% for CIDDS-002, 97.00% for ISCX-Tor2016, 96.11% for UNSW-NB15, 99.86% for CIC-IDS17, 97.76% for ISCX-URL16, 99.91% for BoT-IoT, and 97.43% for the IoTID2020 and Kyoto 2006+ datasets, respectively. The training lasts for 1.10 to 21.78 seconds. The model's efficacy confirms that it can be applied in real-time to any industrial network traffic, including fog cloud computing and intelligent Internet of others settings, [15].

3 Intrusion Detection System

Intrusion detection systems, which are an essential branch of cybersecurity, use different approaches to scan an internet network for illegal activities or attacks and monitor the analysis of network traffic. They are both under the two major categories of Signature-based systems and Anomaly-based systems, [16]. An intrusion detection system is a common cybersecurity tool that gathers, processes, and analyzes data from either connections or computing hosts to identify malicious activity and breaches in security, such as attacks from within or outside the infrastructure, [17]. To create a baseline model, Anomaly-based systems operate by modelling the typical behavior of the system and comparing it to the observed behavior, [18]. Standardly, intrusion detection systems are identified as cybersecurity tools that gather, process, and analyze data from computer hosts or networks to detect breaches in security “attacks that occurred inside or outside of the infrastructure" and other malicious behavior, [19].

4 Dataset

This paper is based on the RT-IoT2022 dataset, which was created from real-time IoT infrastructure, i.e., real network traffic in an IoT environment. It forms an all-inclusive dataset that has a diverse range of IoT devices along with sophisticated

network attack vectors to fully encompass the effort of contributing as much as an authentic dataset that uses real IoT devices to train and evaluate systems for intrusion detection systems (IDSs) in IoT platforms. The dataset contains values returned from IoT devices such as Thing Speak-LED, Wipro-Bulb, and MQTT-Temp, and showed simulated attack vectors that included SSH brute-force attempts, and DDoS attack drivers through Hping, Slowloris, and Nmap type patterns to incorporate benign and adversarial behaviors in the same dataset. While this is only an example set of behaviors, the RT-IoT2022 dataset will also provide a valuable perspective on the potential complex behavior of network traffic. The Zeek network monitoring tool and flow sensor extension are used to record the bidirectional characteristics of network traffic accurately. To enhance the performance of intrusion detection systems (IDSs) and facilitate the development of resilient and robust security solutions for real-time IoT networks, researchers can utilize the RT-IoT2022 dataset.

This dataset was later donated to the UCI Machine Learning Repository, making it available to cybersecurity and IoT researchers. A small CSV file containing a total of 123117 records and 49 features (Table 1) was used. Class labels were provided so that binary and multi-class classification of network activity could be performed using supervised learning. Table 1 shows the class labels so that binary and multi-class classification of network activity can be performed using supervised learning.

Table 1. RT-IoT2022 Dataset Features

Variable Name	Role	Type	Description	Units	Missing Values
id.orig_p	Feature	Integer			no
id.resp_p	Feature	Integer			no
proto	Feature	Categorical			no
service	Feature	Continuous			no
flow_duration	Feature	Continuous			no
fwd_pkts_tot	Feature	Integer			no
bwd_pkts_tot	Feature	Integer			no
fwd_data_pkts_tot	Feature	Integer			no
bwd_data_pkts_tot	Feature	Integer			no
fwd_pkts_per_sec	Feature	Continuous			no

Source: created by author

4.1 Preprocessing

The complete RT-IoT2022 dataset was uploaded for preprocessing. The raw RT-IoT2022 dataset is scrutinized to eliminate errors like missing, incomplete, and inconsistent values. Additionally, duplicates were removed from both rows and columns. The tag encoder replaces categorical features in a dataset containing the attack category, all TCP and IP tags, and protocol type with numeric values for future processing. Specific features such as the source IP address, destination IP address, and timestamp are also removed from the RT-IoT2022 dataset, which are not necessary for detecting attacks.

4.2 RT-IoT2022 Attack Features Analysis and Distribution

The attack types and their distribution ratios were analyzed, primarily by calculating and visualizing the distribution of values for the 'Attack type' column and checking whether the column named 'Attack type' is present in the data frame to count the occurrences of each unique value in the 'Attack type' column. The result (attack counts) is a Pandas string where the index represents the attack types and the values represent the number of occurrences of each type. This is an important step to avoid errors if the column is missing. The data is visualized through a chart with the x-axis representing the attack types and the y-axis representing the number of occurrences of each attack type. Figure 1 shows a visual representation of the percentages of attack types.

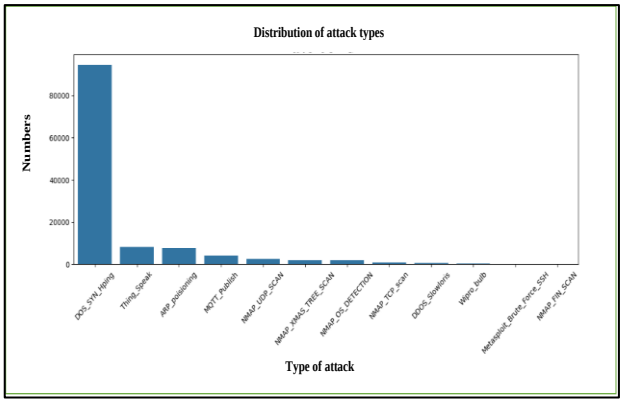

Fig. 1: Types of attacks
Source: created by author

4.3 RT-IoT2022 Attack Features Distribution

The linear relationships between the numeric variables in the data frame were analyzed and visualized. First, columns containing numeric data were selected, and a correlation matrix was calculated between these numeric columns. The correlation coefficients between each pair of variables are shown in this matrix; values near 1 stand for a strong positive correlation, values near -1 for a strong negative correlation, and values near 0 for a weak or nonexistent association. This helps detect potential multicollinearity and understand the

structure of relationships between numeric attributes in the data. Figure 2 shows the confusion matrix for numeric attributes using a heat map.

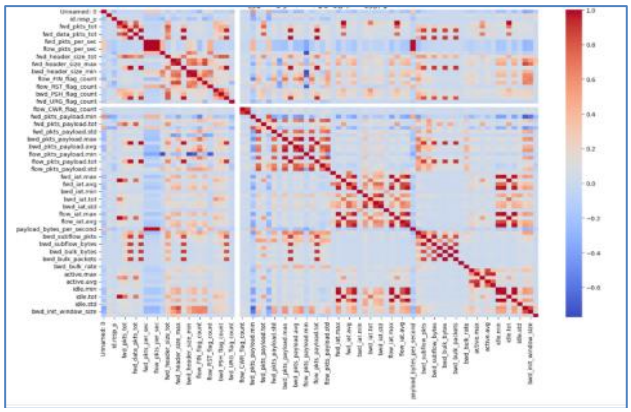

Fig. 2: Correlation map between digital features
Source: created by author

4.4 CGAN (Network of Generative Adversaries)

CGAN architecture was proposed which is made up of two parts, known as the Generator and Discriminator. To In simple terms, the purpose of the generator is to generate new data (like numbers or images) that matches closely with the input dataset, at the same time, the purpose of the discriminator is to identify generated data versus actual input data in Figure 3. A multi-layer perceptron (MLP) neural network was used in this CGAN for both the discriminator and generator, [20].

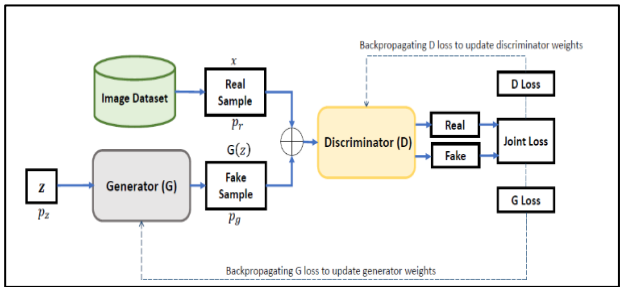

Fig. 3: Generative Adversarial Networks – GANs
Source: [19]

5 Machine Learning Integrated Intrusion Detection Systems

Artificial intelligence (AI) and machine learning (ML) have radically changed the domain of commercial security by enhancing intrusion detection techniques in modern security systems. Before we can explore the importance and potential of AI-based intrusion detection, we need to describe the algorithms and techniques that are used to underpin these systems. Machine learning is a central capability for intrusion detection systems (IDSs) that enables the capacity to identify and respond to threats to security. Intrusion detection systems (IDS) and security information event management systems (SIEMs) utilize machine learning models to analyze network and systems performance, traffic, and activity patterns to identify anomalous activity that may signify an attack (True Protection, 2024). Machine learning algorithms are trained on sizable data sets to recognize complex patterns that the human eye may have difficulty identifying. One common algorithm used in IDSs is the LightGBM algorithm, which is not a standard algorithm and is widely known for its specific set of equations in the field of intrusion detection. We can infer its potential mathematical structure as a lightweight gradient boosting model designed to operate efficiently in IDSs.

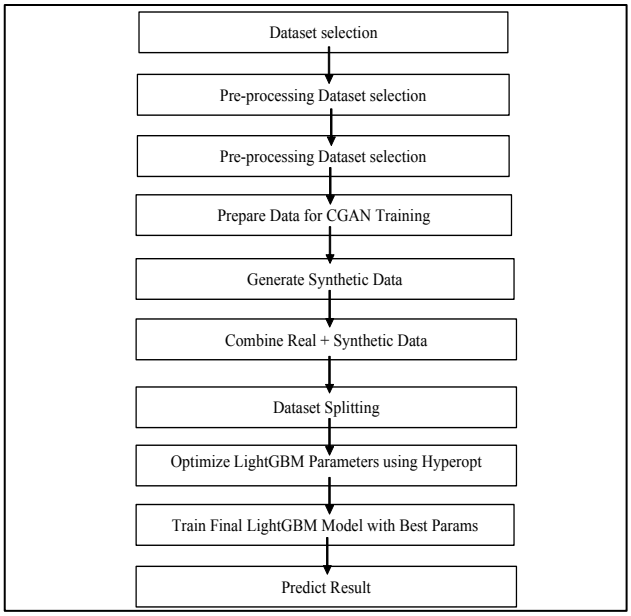

Fig. 4: Flowchart of Proposed Methodology
Source: created by author

Figure 4 shows the main utilized architecture staring from data selection, process it to machine understandable format and prepare it for training. And then evaluate on unseen data.

5.1 Generators

The model's Generator Network is the more intricate of the two. Using a random seed, often known as a noise sample, to train a "vanilla" GAN (the original Goodfellow model), the generator immediately begins generating samples. The initial endeavors yield poor results since they consist mainly of random noise. However, as the discriminator sends more feedback to the generator, the latter gradually

enhances the quality of its samples, aligning them more closely with those in the training set. After convergence is achieved or the full number of training epochs has run, the Generator is typically the part of the model that is retained. When the training is finished and the generator can create samples that are nearly indistinguishable from real ones, it is prepared to be employed for its intended use.

5.2 Discriminators

Usually, after the generator has been effectively trained, the discriminator in a model is removed. Analyzing and correctly classifying the samples provided by the generator, regardless of their legitimacy, is the fundamental job of a discriminator. Over time, the discriminator should grow less effective, eventually resembling a computer lottery, as the generator receives feedback and gradually modifies its weights to create more accurate samples. Sometimes, the tool that discriminates wins the game, and depending on the kind of GAN model, this can produce an accurate and useful classifier.

5.3 Loss Functions

The training process involves a minimax game with the following loss functions: Discriminator Loss (LD):

$$LD(\theta D, \theta G) = Ex \sim pdata(x), y \sim pdata(y)[logD(x,y)] + Ez \sim pz(z), y \sim pdata(y)[log(1 - D(G(z,y;\theta G),y))]$$

The result of the discriminator function D (x; θD) is a likelihood value (just one integer), which shows whether the input x originated from either the generator or the training set.

$$minmaxV(D,G) = E_{X\sim pdata(x)}[log\ D(x)] \quad (1)$$

$$E_{z\sim pz(z)}[log(1 - D(G(z)))]$$

Equation (1), this is the expected value (E) of the logarithm (log) of the probability that the Discriminator (D) assigns to a real data sample (x) being real. Equation (2), this indicates that the generator tries to fool the discriminator into believing that the generated data G(z,y), conditioned on label y, is real and belongs to that classy.

5.4 Generating Attack Data and the Models Definition using a GAN

At the beginning of model building, additional attack data is generated using GANM to learn the characteristics of real attacks and produce similar data. The generated data is preprocessed through a batch normalization routine, which comes after a series of linear layers in the model. This is followed by a distribution intended to distinguish between the actual data and the data generated by the generator. This model also consists of a series of linear layers, but the Leaky ReLU activation function and dropout layers are used to help prevent overfitting and improve stability during training. The network terminates with a single linear layer, which produces a value between 0 and 1 representing the probability that the input data is real and not generated. The routing function in both models determines how data flows through the model layers.

5.5 Training a Conditional Generative Adversarial Network (CGAN) Model

The GANs can be used to generate various types of attack data, including: First, network intrusion data, where GANs can generate network traffic data simulating various types of network attacks to train intrusion detection systems (IDSs). Second, malware samples, where GANs can generate synthetic malware samples exhibiting characteristics similar to real malware to train anti-malware engines and develop new detection techniques. Third, vulnerability exploitation data, where GANs can generate data simulating exploit attempts. Figure 5 illustrates the loss function and model training on the generated attack data. To evaluate the proposed LightGBM model, the collected data, with randomly selected characteristics, was divided into two sets: training (80%) and testing (20%). Machine learning (ML) offers a feasible solution to these challenges.

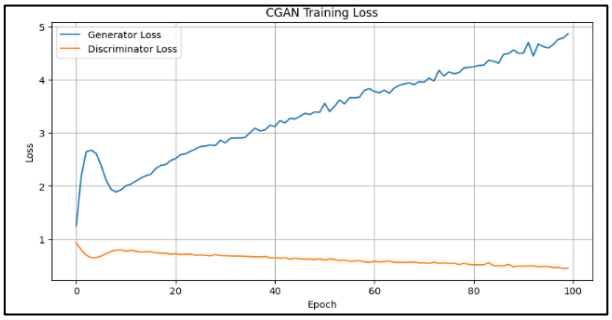

Fig. 5: Illustrates the Loss Function and the Model's Training on the Generated Data
Source: created by author

5.6 Generating and Displaying Synthetic Data Similar to the Original Data

5.6.1 Data-Generating Models

The model responsible for generating data is the generator, which was trained as part of a Generative Adversarial Network (GAN). This model is trained

within the GAN architecture to transform random noise into synthetic data, attempting to mimic the distribution of real data. The generate synthetic data function is used to generate data and then reverses the scaling process to return the data to its original scale. This makes the synthetic data more realistic and comparable.

5.6.2 Objective Function of GANs

The Objective Function of GANs Objective function of an artificial neural network is the difference between the probability distribution of generated samples (pg) and the distribution of real samples (pr). We can calculate the objective function by leveraging binary cross-entropy loss. a joint loss function for the discriminator and generator is binary cross-entropy V (D, G): It reduces the Jensen-Shannon divergence (JSD) between the generated and real data distributions.

5.7 Applications of Gans in Detection of Intrusion Attacks

Generative adversarial networks (GANs) have been implemented in inventive ways within the intrusion detection domain to augment intrusion detection systems (IDSs). Researchers recently proposed an IDS approach reliant on a Conditional Tabular GAN (CTGAN), which aims to address class imbalance in IoT networks through adding synthetic data for infrequent intrusions, [21]. In contrast, a different research article from 2024 sought to explore how GANs could improve the intrusion detection capability of networks by creating different (but accessible) attack samples that would allow for better generalization on unfamiliar attacks. Furthermore, one notable MDPI article from 2023 reported the work of researchers who applied GANs to generate adversarial samples for the purpose of evaluating IDS vulnerabilities during adversarial training to optimize robustness, [22]. The use of GANs has extended to specific environments such as SCADA systems, where GAN-based network intrusion detection systems have been developed and achieved high accuracy in anomaly detection. Study conducted a systematic investigation of cloud data leakage, security, privacy issues, and related challenges, emphasizing the critical need for robust mechanisms to address vulnerabilities in cloud environments, [23]. Study on applied the “Apriori association rule algorithm in a cloud environment” to enhance intelligent transmission line fault diagnosis, enabling scalable and accurate analysis for smart grid monitoring, [24].

6 Evaluation Metrics

The following metrics were used in order to guarantee a comprehensive evaluation:

- Accuracy: The general correctness of the predictions made by the model.
- Precision: The percentage of actual positive forecasts out of all positive ones.
- Recall: The percentage of all true positives that are true positives.
- F1-score: A harmonic representation of recall and precision.

6.1 Classification of Performance

The LightGBM model's classification of performance for traffic-level anomaly detection was evaluated. Figure 6. Illustrate the flowchart of the proposed methodology.

The confusion matrix provides a visual illustration of the prediction performance. Figure 6 shows the LightGBM model evaluation report.

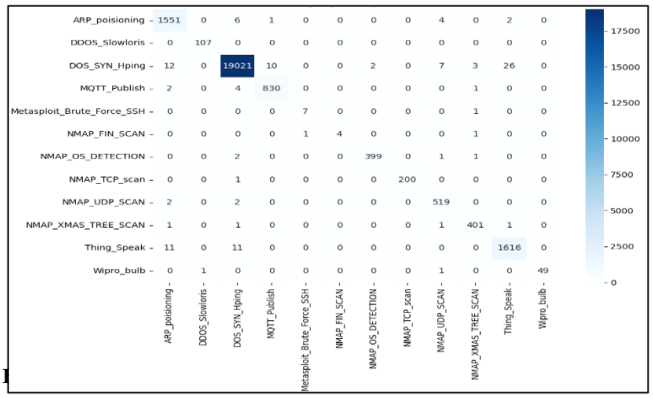

Fig. 6: Confusion Matrix for Light GBM Model
Source: created by author

6.1 Feature Extraction

After conducting data preprocessing and filtering, relevant features were chosen and identified from the raw data to improve upon the predictive performance of the model that classifies both malicious and benign attacks. This process is an important step; that is because the quality and importance of features play a role in determining the effectiveness and efficiency of a machine learning model. This section contains a visual representation of the Receiver Operating Characteristic (ROC) curve in Figure 7.

The receiver operating characteristic (ROC) curve is a graphical representation of a binary classification model's performance at different classification thresholds, and a tool for evaluating the performance of the model.

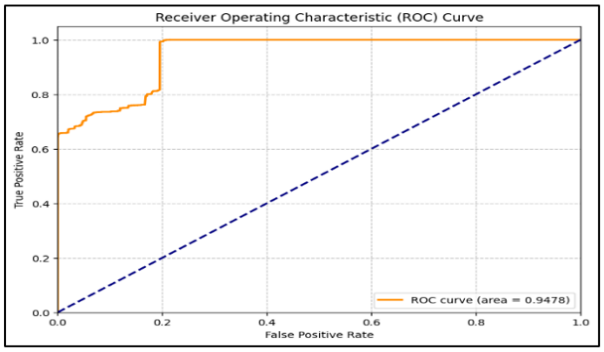

Fig. 7: Receiver Operating Characteristic (ROC) Curve
Source: created by author

The AUC value indicates the excellent ability of the model to discriminate between two groups, at a very high level of .9478. A high AUC value means that the model was able to maintain low false positives while advising high true positives. The classification report is showing in Table 2.

Table 2. Show the results based on evaluation metrics.

	precision	recall	F1-score	support
Normal	0.99	0.80	0.89	439683
Attack	0.74	0.99	0.85	251723
accuracy			0.87	691406
macro avg	0.87	0.90	0.87	691406
weighted avg	0.89	0.87	0.88	691406

Source: created by author

The results of the evaluation of a binary classification model appear in the Table 2. The objective was to separate normal from attack traffic. The top portion of the table represents the confusion matrix, which describes the performance of the model by jointly displaying actual classifications and predicted classifications. The confusion matrix indicates that the model correctly classified 438,627 normal instances, but mistakenly classified 1,056 normal instances as attacks. For attacks, the model correctly classified 165,114 attacks, but mistakenly classified 86,609 instances as normal.

6.3 The Importance of Features in the Model

This chart represents a horizontal bar showing the importance of the first 20 features in a trained machine learning model. The assessment of feature importance provides an understanding of the features in network data that contributed most to the differentiation between normal and malicious traffic for that specific model. This can provide fundamental insights to improve the model, understand how the networks behave, or even inform security strategies that may put a greater emphasis on these important features. Figure 8. provides information on 20 feature importance in the proposed model.

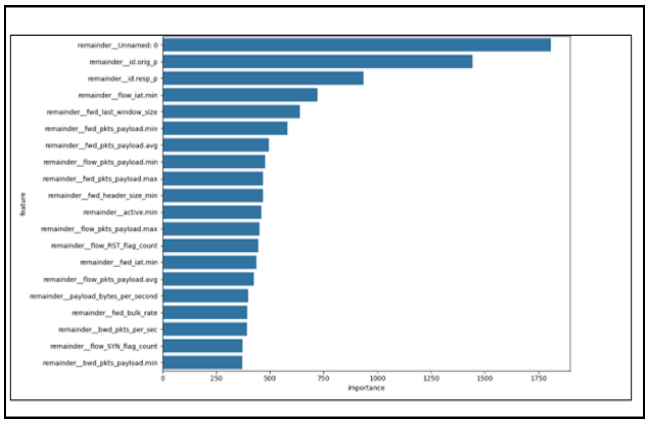

Fig. 8: The Importance of 20 Features in the Proposed Model
Source: created by author

3 Discussion and Results

In order to check the proposed intrusion detection model based on these results, accuracy conveys overall model performance but can be misleading on imbalanced datasets. The training and testing data suggests that it reflects realistic situations, is clean, and is not significantly biased and that the way the data was split into training and testing data was done correctly, in random ways so no data appears to have leaked. The confusion matrix provides details at a class level for the number of correct and incorrect classifications in each class and helps us understand what kinds of errors the model is making. The model appears good at identifying normal cases and does misclassify a large number of attacks as Overall, while the model shows high accuracy and pretty good accuracy for predicted attacks, the relatively low recall for attacks raises concerns about whether it can detect all of the real attacks. Complete validation involves assessing all metrics, understanding the context of the application (what is the cost of missing an attack compared to false alarms?), comparative performance with other models, and testing on new data. Based on the code output, the results of the evaluated model can be summarized in Table 3.

Table 3. The Result of Accuracy metrics

Accuracy:	(87.32%)
Precision for Attack:	(99.36%)
Recall for Attack:	(65.59%)
F1 Score for Attack:	(79.02%)
Area under the ROC Curve (AUC):	(94.78%)

Source: created by author

For the "Normal" class, the model achieved a precision of 0.84, a recall of 1.00, and an F1 of 0.91. For the "Attack" class, it achieved a high precision of 0.99, but a relatively lower recall of 0.66, resulting in an F1 of 0.79. The high precision for the "Attack" class indicates that the model is very accurate when predicting an attack, but the low recall indicates that it misses a large number of actual attacks.

3.1 Analysis of the Confusion Matrix Table and Classification Report

The table shows the performance evaluation results of a binary classification model. The first section is the confusion matrix, which compares the actual classifications to the classifications predicted by the model. The second section is the classification explanation, which provides a more granular view of performance metrics for each class (Normal and Attack) in addition to the overall metrics. Table 4. shows the counts of normal and attack predictions.

Table 4. Number of normal and attack predictions

Class	**Predicted Attack**	**Predicted Normal**
Actual Normal	438627	1056
Actual Attack	86609	165114

Source: created by author

4 Conclusion

This method describes the inclusion of GANs into the Light model framework for a machine learning task to address the important issue of low data rates seen on IDS training datasets. The method describes the development and use of a novel GAN model to generate synthetic data of network traffic that closely represents the real network traffic. Many experiments were performed on both a benchmark dataset (RT-IoT2022) and test beds using the generated datasets. The experiments showed clear evidence that utilizing GANs in the Light model improved intrusion detection.

A machine learning model for network intrusion detection, called the LightGBM model, was evaluated. Last, the evaluation presented the top 10 features of the data that spurred the model's predictions, thus providing a more informative understanding of the most indicator features of intrusion in the data. The use of many different metrics also enabled an overall evaluation across the model.

The evaluation results show that the model produced an accuracy of 0.8732. Additionally, it produced high attack precision (0.9936) meaning that when the model predicts an attack, it is generally reliable. However, the recall of attacks was average (0.6559) meaning that the model misses some actual attacks. The F1 measure of attacks (0.7902) is consistent with the moderate trade-off between precision and recall. The confusion matrix also indicates that the model is good at classifying normal cases while it also produces a large number of false negatives. Finally, the AUC value of 0.9478 indicates good discriminatory ability between normal and malicious traffic.

Declaration of Generative AI and AI-assisted Technologies in the Writing Process

The author wrote, reviewed and edited the content as needed and verifies that none utilized artificial intelligence (AI) tools were used. The author takes full responsibility for the content of the publication.

References:

[1] B. R. Kikissagbe and M. Adda, "Machine Learning-Based Intrusion Detection Methods in IoT Systems: A Comprehensive Review," *Electronics,* vol. 13, no. 18, art.no.3601,2024,doi: https://doi.org/10.3390/electronics13183601.

[2] B. C. Manjith and N. Ramasubramanian, "AES hardware accelerator on FPGA with improved throughput and resource efficiency," *Arab. J. Sci. Eng.,* vol. 43, no. 12, pp. 7331–7341, 2018, doi: 10.1007/s13369-017-2925-0.

[3] H. Iguer, H. Medromi, A. Sayouti, S. Elhasnaoui, and S. Faris, "The Impact of Cyber Security Issues on Businesses and Governments: A Framework for Implementing a Cyber Security Plan," in 2014 *International Conference on Future Internet of Things and Cloud*, 2014, pp. 316–321. doi: 10.1109/FiCloud.2014.56.

[4] G. Karatas, O. Demir, and O. Sahingoz, "Deep Learning in Intrusion Detection Systems Gozde," *2018, Ankara, Turkey*. doi: 10.1109/IBIGDELFT.2018.8625278.

[5] M. Mahmoud, Y. Youssef, and A. Abdel-Hamid, "XI2S-IDS: An Explainable Intelligent 2-Stage Intrusion Detection System," *MDPI*, vol. 17, p. 28, 2025, doi: 10.3390/fi17010025.

[6] S. Tripathy and D. B. Behera, "A Review of Various Datasets for Machine Learning Algorithm-Based Intrusion Detection System: Advances and Challenges," *Intell. Syst. Appl. Eng.*, no. 2147–67992, p. 26, 2024.

[7] O. Ogundairo and P. Broklyn, "Machine Learning Algorithms for Intrusion Detection Systems," *J. Cyber Secur.*, 2024.

[8] I. Martins, J. Resende, P. Sousa, S. Silva, L. Antunes, and J. Gama, "Host-based IDS: a review and open issues of an anomaly detection system in IoT," *Future Generation Computer Systems*, Vol. 133, August 2022, pp.95-113 2022. https://doi.org/10.1016/j.future.2022.03.001.

[9] A. Ferdowsi and W. Saad, "Generative Adversarial Networks for Distributed Intrusion Detection in the Internet of Things," *2019 IEEE Global Communications Conference (GLOBECOM)*, Waikoloa, HI, USA, 2019. https://doi.org/10.1109/GLOBECOM38437.2019.9014102.

[10] M. Ferrag, D. Hamouda, M. Debbah, L. Maglaras, and A. Lakas, "Generative Adversarial Networks-Driven Cyber Threat Intelligence Detection Framework for Securing Internet of Things," *2023 19th International Conference on Distributed Computing in Smart Systems and the Internet of Things (DCOSS-IoT)*, Pafos, Cyprus, 2023. https://doi.org/10.1109/DCOSS-IoT58021.2023.00042.

[11] B. Alabsi, M. Anbar, and S. Rihan, "Conditional Tabular Generative Adversarial Based Intrusion Detection System for Detecting Ddos and Dos Attacks on the Internet of Things Networks," *MDPI*, vol. 23, p.20, 2023, doi: 10.3390/s23125644.

[12] S. Li, Y. Cao, S. Liu, Y. Lai, Y. Zhu, and N. Ahmad, "HDA-IDS: A Hybrid DoS Attacks Intrusion Detection System for IoT by using semi-supervised CL-GAN," *Expert Syst. Appl.*, vol. 238, p. 122198, 2024, https://doi.org/10.1016/j.eswa.2023.122198.

[13] K. Singal, N. Kandhoul, and S. K. Dhurander, "Evolutionary LightGBM-Based Intrusion Detection System for IoT Networks," *Int. J. Commun. Syst.*, vol. 38, no. 5, p. e70031, Mar. 2025, https://doi.org/10.1002/dac.70031.

[14] Y.-M. Yang, K.-C. Chang, and J.-N. Luo, "Hybrid Neural Network-Based Intrusion Detection System: Leveraging LightGBM and MobileNetV2 for IoT Security," *MDPI*, vol. 17, p. 19, 2025, doi: 10.3390/sym17030314.

[15] S. A. H. Ayubkhan, W.-S. Yap, E. Morris, and M. B. K. Rawthar, "A practical intrusion detection system based on denoising autoencoder and LightGBM classifier with improved detection performance," *J. Ambient Intell. Humaniz. Comput.*, vol. 14, no. 6, pp. 7427–7452, 2023, doi: 10.1007/s12652-022-04449-w.

[16] O. H. Abdulganiyu, T. Ait Tchakoucht, and Y. K. Saheed, "A systematic literature review for network intrusion detection system (IDS)," *Int. J. Inf. Secur.*, vol. 22, no. 5, pp. 1125–1162, 2023, doi: 10.1007/s10207-023-00682-2.

[17] A. Milenkoski, M. Vieira, S. Kounev, A. Avritzer, and B. D. Payne, "Evaluating Computer Intrusion Detection Systems: A Survey of Common Practices," *ACM Comput. Surv.*, vol. 48, no. 1, Sep. 2015, doi: 10.1145/2808691.

[18] J. I. I. Araya and H. Rifà-Pous, "Anomaly-based cyberattacks detection for smart homes: A systematic literature review," *Internet of Things*, vol. 22, p. 100792, 2023, https://doi.org/10.1016/j.iot.2023.100792.

[19] M. Saad, R. O'Reilly, and M. Rehmani, "A Survey on Training Challenges in Generative Adversarial Networks for Biomedical Image Analysis," 2022. doi: 10.48550/arXiv.2201.07646.

[20] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, and Y. Bengio, "Generative Adversarial Networks," *Commun. ACM*, vol. 63, p. 6, 2020, doi: 10.1145/3422622.

[21] S. Aldhaheri and A. Alhuzali, "SGAN-IDS: Self-Attention-Based Generative Adversarial Network against Intrusion Detection Systems," *MDPI*, p. 18, 2023.

[22] X. Zhao∗, K. Fok, and V. Thing, "Enhancing Network Intrusion Detection Performance using Generative Adversarial Networks Xinxing," *ST Eng. Singapore*. arXiv2404.07464v1, p.12, 2024.

[23] Salih, B. M., & Mohammad, O. K. J. (2024). Cloud Data Leakage, Security, Privacy Issues and Challenges. *Procedia Computer Science*, 242, p.592-601.

[24] Al-Jumaili, A. H. A., Muniyandi, R. C., Hasan, M. K., Singh, M. J., & Paw, J. K. S. (2023). Intelligent transmission line fault diagnosis using the Apriori associated rule algorithm under cloud computing environment. *All Open Access*. https://doi.org/10.32629/jai.v6i1.640.

Contribution of Individual Authors to the Creation of a Scientific Article (Ghostwriting Policy)

Amjad Jumaah Frhan, carried out the entire research, including problem formulation, methodology design, implementation, experiments, data analysis, and writing of the manuscript.

Sources of Funding for Research Presented in a Scientific Article or Scientific Article Itself

No funding was received for conducting this study.

Conflict of Interest

The author has no conflicts of interest to declare that are relevant to the content of this article.