Journal of Universal Computer Science, vol. 32, no. 8 (2026), 1223-1257
submitted: 19/8/2025, accepted: 29/12/2025, appeared: 28/8/2026 CC BY 4.0

Enhancing IoT Attack Detection Through SMOTE-Based Class Balancing: A Comparative Analysis of Machine Learning Algorithms

Ela Kalifati
(Department of Software Engineering, Kocaeli University, Kocaeli, Türkiye
https://orcid.org/0009-0002-8335-4287, 225129019@kocaeli.edu.tr)

Kerem Küçük
(Department of Software Engineering, Kocaeli University, Kocaeli, Türkiye
https://orcid.org/0000-0002-2621-634X, kkucuk@kocaeli.edu.tr)

Şevval Şolpan
(Department of Software Engineering, Kocaeli University, Kocaeli, Türkiye
https://orcid.org/0000-0002-8484-5697, sevval.solpan@kocaeli.edu.tr)

Kazım Kıvanç Eren
(Department of Computer Engineering, Kocaeli University, Kocaeli, Türkiye
https://orcid.org/0000-0002-5071-2722, kvnceren@gmail.com)

Mehmet Zeki Konyar*
(Department of Software Engineering, Kocaeli University, Kocaeli, Türkiye
https://orcid.org/0000-0001-8914-5553, mzeki.konyar@kocaeli.edu.tr)

Abstract: Imbalanced class distributions in Internet of Things (IoT) attack datasets limit the ability of machine learning (ML) models to detect minority intrusions and lead to high false-negative rates. This study investigates the effectiveness of the Synthetic Minority Oversampling Technique (SMOTE) in improving multi-class attack detection across seven IoT devices from the TON_IoT dataset. Four ML algorithms —K-Nearest Neighbor, Support Vector Machine, Logistic Regression, and Multi-Layer Perceptron— are evaluated both with and without SMOTE using accuracy, precision, recall, and F1-score.

Results show that SMOTE substantially improves minority-class detection, particularly in devices with severe imbalance. On the Garage Door dataset, KNN accuracy increases from 0.952 to 0.982, and MLP from 0.555 to 0.978, while SVM exhibites modest gains. The largest recall improvement is observed for Modbus with KNN, rising from 0.52 to 0.84.

Overall, SMOTE enhances IoT intrusion detection without degrading majority-class performance, although its impact varies across algorithms and device characteristics. Future research is planned to explore hybrid resampling and deep learning-based approaches to further improved detection under extreme imbalance.

Keywords: Internet of Things, Data Imbalance, Synthetic Minority Over-sampling Technique, Machine Learning
Categories: I, I.2
DOI: 10.3897/jucs.169225

1 Introduction

The rapid evolution of digital technologies has fundamentally reshaped global socioeconomic systems, ushering in a new era of interconnectivity. Central to this transformation is the Internet of Things (IoT)—a network of physical devices ranging from household appliances to industrial sensors that autonomously exchange data and enable intelligent decision-making. By linking digital and physical infrastructures, IoT enhances operational efficiency, predictive analytics, and sustainable resource management. Its widespread adoption represents not just an advancement in automation, but a structural shift in human-environment interactions, as discussed by Ait Mouha [Ait Mouha, 21]. Despite its vast potential, IoT implementation faces persistent challenges, including interoperability issues among heterogeneous devices, exposure to cyber threats, and complexities in large-scale data governance. These obstacles hinder the reliability and scalability of IoT ecosystems, underscoring the need for robust and adaptive solutions, as reported by Gokhale et al. [Gokhale et al., 18; Mazhar et al., 23]. A critical issue in IoT-based cybersecurity is the imbalance of data distributions generated by connected devices. This uneven representation of normal and malicious activity leads to biased training of machine learning (ML) models, which often misclassify rare yet significant events as normal traffic. Consequently, anomaly detection systems suffer from high false-negative rates, undermining network integrity and exposing IoT environments to sophisticated cyberattacks, as discussed by Farea et al. [Farea et al., 24] and Eren and Küçük [Eren and Küçük, 23].

While ML has emerged as a powerful tool to analyze and classify IoT data, the effectiveness of these algorithms is often limited by imbalanced datasets. Advanced ML algorithms —such as deep learning (DL) and ensemble techniques— demonstrate strong capabilities in processing diverse IoT data streams; however, class imbalance biases their decision boundaries toward majority classes, reducing detection accuracy for critical minority events, as discussed by Hussain et al. [Hussain et al., 20] and Adi et al. [Adi et al., 20].

To mitigate this problem, resampling-based approaches such as the Synthetic Minority Oversampling Technique (SMOTE) have proven effective. SMOTE generates synthetic minority samples through linear interpolation between existing instances and their nearest neighbors, thereby enhancing minority class representation without distorting the original data distribution. This leads to more balanced training, improved model generalization, and stronger anomaly detection in security-critical IoT applications, as demonstrated by Mohindru et al. [Mohindru et al., 21], Eid et al. [Eid et al., 24], Pradipta et al. [Pradipta et al., 21], and Soe et al. [Soe et al., 19].

The objective of this study is to address the class imbalance problem in IoT datasets and evaluate the effectiveness of SMOTE as a balancing strategy. It investigates the effectiveness of the SMOTE technique by improving ML model performance on imbalanced IoT intrusion detection data from the ToN_IoT dataset. It compares K-Nearest Neighbor (KNN), Support Vector Machine (SVM), Logistic Regression (LR), and Multi-Layer Perceptron (MLP) models on both original and balanced with SMOTE data to analyze how oversampling affects multi-class classification accuracy across different IoT devices. The main contributions of this work are summarized as follows:

• Investigation of the impact of class imbalance on multi-class IoT intrusion detection performance.

• Comparative evaluation of four ML algorithms (KNN, SVM, Logistic Regression, and MLP) both with and without SMOTE.
• Analysis of how SMOTE influences accuracy, precision, recall, and F1-score across seven different IoT device datasets.
• Proposal of more balanced and effective preprocessing strategies for IoT security applications.

The structure of this study is organized as follows. Section 2 presents the related works. Section 3 describes the dataset, preprocessing pipeline, classification algorithms, and SMOTE methodology. Section 4 provides detailed performance analysis for two representative IoT devices. Section 5 extends the comparison across all devices using aggregate performance tables. Section 6 discusses the key findings and implications, concludes the study and outlines directions for future research.

2 Related Works

2.1 Data-Level Approaches for Imbalanced Data

The SMOTE is used by Mohammed et al. [Mohammed et al., 20] to enhance the classification performance of an unbalanced medical dataset. Their findings show that resampling significantly increases model accuracy, demonstrating SMOTE's usefulness in balancing datasets. Additionally, the results demonstrate that SMOTE enhances the resampling rate and permits more equitable results in classification algorithms. Blagus and Lusa [Blagus and Lusa, 13] examine how SMOTE affects unbalanced and high-dimensional datasets, including gene expression data. They compare SMOTE's performance on high-dimensional datasets with other approaches. The findings demonstrate that SMOTE works well for low-dimensional datasets but offers few advantages for high-dimensional data. In particular, they discover that when paired with variable selection methods, SMOTE enhances the performance of classifiers such as KNN. They stress, however, that straightforward undersampling techniques are typically more successful for high-dimensional datasets and frequently result in increased classifier accuracy.

Tasnim et al. [Tasnim et al., 23] conduct a systematic evaluation of six ML and DL classifiers —Decision Tree (DT), Random Forest (RF), AdaBoost, XGBoost, Artificial Neural Network (ANN), and MLP— on the TON_IoT dataset for IoT intrusion detection. The study examines the impact of data imbalance by applying three balancing techniques: undersampling, oversampling, and SMOTE. Results indicate that DT and RF consistently achieve the highest performance, exceeding 99% accuracy in both binary and multi-class scenarios, while AdaBoost and ANN underperformed in multi-class classification. This work highlights the importance of model selection and appropriate handling of imbalanced IoT datasets for effective intrusion detection.

2.2 Machine Learning for IoT Intrusion Detection

ML, thanks to its ability to automatically learn from data and make decisions, has become one of today's most critical technologies. By minimizing human intervention, it extracts meaningful insights from complex data, reduces errors, and optimizes processes. By offering innovative solutions in numerous fields, including healthcare, as demonstrated by Al-Areqi and Konyar [Al-Areqi and Konyar, 22], finance, as discussed

by Kocaoğlu et al. [Kocaoğlu et al., 22], manufacturing, as investigated by Saleh et al. [Saleh et al., 24], and predictive maintenance, as explored by Öngir et al. [Öngir et al., 25], it not only increases efficiency but also adds speed and accuracy to strategic decision-making processes.

The preliminary result for this paper obtanied with our previous conference proceeding Kalifati et al. [Kalifati et al., 25] investigated the effect of SMOTE on the TON_IoT dataset using KNN and SVM across seven IoT devices.

In order to identify cyberattacks that target network devices in Internet of Things environments, Calp and Bütüner [Calp and Bütüner, 24] offer a model that makes use of ML methods including KNN, RF, and ANN. According to their study's findings, the RF algorithm performs exceptionally well, with the greatest accuracy of 99.6%. This result implies that the RF algorithm is a useful technique for identifying cyberattacks in Internet of Things systems.

To improve security solutions for vehicle networks, Gad et al. [Gad et al., 21] create an intrusion detection system (IDS). They try different ML algorithms and use SMOTE and the Chi-square test for feature selection in the creation of this system. The XGBoost algorithm performs better than the others, as seen by the results.

Verma and Ranga [Verma and Ranga, 20] create intrusion detection systems for the Internet of Things by assessing the effectiveness of several machine learning techniques, such as RF, AdaBoost, Gradient Boosting Machine (GBM), and XGBoost. According to their findings, XGBoost surpasses the other algorithms in terms of AUC (Area Under the Curve), demonstrating its excellent capacity to distinguish between various classes, whereas RF achieves the highest accuracy among the evaluated methods.

Alotaibi and Ilyas [Alotaibi and Ilyas, 23] propose an ensemble-based IDS for multi-class attack classification in IoT networks using the TON_IoT dataset. After evaluating five high-performing base models (RF, DT, Extra Tree, XGBoost, KNN), they implement stacking and voting ensemble techniques. The ensembles significantly enhance the classification performance, with the voting ensemble achieving the highest accuracy of 98.9% and near-perfect scores across the majority of attack categories. This study underscores the efficacy of ensemble learning in bolstering the robustness and accuracy of IoT intrusion detection systems.

2.3 Hybrid and Advanced Approach

The SMOTE-Deep Recurrent Neural Network (DRNN) algorithm was developed by Popoola et al. [Popoola et al., 21] to identify botnet attacks in Internet of Things environments. The purpose of this approach is to improve training datasets' imbalanced network traffic data classification accuracy. The findings show that the SMOTE-DRNN model performs better in terms of accuracy, precision, and F1-score than other ML and DL models. The model is a dependable method for identifying security risks since it also demonstrates resilience against both overfitting and underfitting.

A simplified ANN is developed by Soe et al. [Soe et al., 19] to identify distributed denial-of-service (DDoS) threats in Internet of Things contexts. They use the Bot-IoT dataset for experiments and the SMOTE approach to address the problem of data imbalance. The model obtains 100% accuracy after using SMOTE, demonstrating its efficacy in identifying these attacks, according to the data.

To solve the classification issue in imbalanced datasets, Zhang et al. [Zhang et al., 22] create a model known as SEB-XGB. This model uses EasyEnsemble to balance the majority class and SVM-SMOTE to strengthen the minority class. The findings of their investigation, which uses Bayesian Optimization for parameter adjustment, demonstrate that SEB-XGB performs better in G-mean and AUC measures. Additionally, the model offers better accuracy rates on credit card and fraud datasets when compared to conventional XGBoost. An efficient classification technique for unbalanced datasets is proposed in this paper.

To find security flaws in IoT networks and improve classification accuracy, Doghramachi and Ameen [Doghramachi and Ameen, 23] provide a method that combines XGBoost with SMOTE. The IoTID20 dataset is used in their investigation, and the model performs well when handling imbalanced data categorization. The model's superiority is demonstrated by the results, which show an accuracy of 0.81 and a recall of 0.83, especially in binary and multi-class classification tasks. Additionally, as compared to earlier research, the suggested approach significantly improves G-mean and AUC metrics. The researchers advise creating more flexible systems to fend off cyberattacks and modeling dynamic IoT ecosystems for future research.

The goal of Eren and Küçük [Eren and Küçük, 23] is to improve the effectiveness of intrusion detection systems for Internet of Things devices. In order to do this, they employ feature preprocessing and automated feature extraction techniques to the TON_IoT dataset, which represents contemporary IoT contexts. The findings demonstrate that the RF algorithm performs exceptionally well in identifying sophisticated attacks, achieving a ROC-AUC score of 99.99%. The study also emphasizes the need for more effective ways to deal with problems caused by scenarios with little data, like MITM assaults. There are still a number of important gaps in the literature despite significant progress in the field. The majority of research ignores the multi-class nature of actual IoT intrusion detection datasets in favor of binary classification. Furthermore, there are few comparison studies evaluating SMOTE's efficacy across various IoT device kinds and threat categories, and its application on extensive datasets such as TON_IoT is restricted. This work studies KNN and SVM performance in various IoT scenarios, applies SMOTE to the multi-class TON_IoT dataset, and examines its effects on accuracy, precision, and F1-score in order to address these problems. These contributions provide an in-depth understanding of how data balancing improves intrusion detection in heterogeneous IoT systems.

Cao et al. [Cao et al., 24] leverages the aggregated TON_IoT dataset, combining device and network data, to replicate complex IoT environments.

A key aspect of their methodology is the application of sine and cosine cyclic encoding to better represent temporal features. To tackle class imbalance, they employ a hybrid resampling technique using SMOTE for oversampling, followed by a refined Tomek-link method for undersampling to eliminate redundancy.

The study evaluates six algorithms —KNN, RF, Classification and Regression Trees (CART), Convolution Neural Network (CNN), Long Short-Term Memory (LSTM), and Deep Neural Networks (DNN)— across six preprocessed datasets. The results demonstrate that their approach achieves a high performance of 98.7–99.4% in multi-class classification after applying SMOTE with Tomek-link removal. This underscores the efficacy of integrating temporal feature engineering with advanced class-balancing techniques to enhance intrusion detection in complex IoT systems.

Venkatasubramanian et al. [Venkatasubramanian et al., 25] propose a comprehensive SMOTE-enhanced Transformer-based model to tackle dataset imbalances and computational challenges in IoT intrusion detection.

Their methodology integrates a novel feature selection technique using the Black-winged Kite Algorithm (BKA) and addresses class imbalance with SMOTE oversampling. The study extensively evaluates four pre-trained Transformer models — BERT, DistilBERT, RoBERTa, and XLNet— on the BoT-IoT, TON_IoT, and CIC-DDoS2019 datasets.

While RoBERTa demonstrates strong contextual representations, the ultimate performance is achieved by an ensemble transformer model, which significantly outperforms all baseline models, including individual transformers and DL models like AutoEncoder (AE), Deep Belief Network (DBN), Gated Recurrent Unit (GRU), Recurrent Neural Network (RNN), and LSTM. The proposed system achieves a remarkable test accuracy of 99.7% with minimal standard deviation, highlighting the efficacy of combining advanced feature selection, data balancing, and ensemble learning with transformer architectures for robust and efficient IoT security.

3 Proposed Methodology

The following procedures are used in this study to analyze data and assess ML models' performance using the TON_IoT dataset. Figure 1 shows the general workflow of these procedures.

3.1 Dataset and Preprocessing

In order to assess cybersecurity risks in IoT and Industrial IoT (IIoT) environments, this study makes use of the TON_IoT dataset, which was created by the Cyber IoT Lab at UNSW Canberra [Moustafa, 21].

The TON_IoT dataset comprises both normal network activity and diverse attack scenarios, providing a robust foundation for evaluating the detection capabilities of ML algorithms in IoT security applications, as described by Alsaedi et al. [Alsaedi et al., 20]. The dataset consists of IoT telemetry records collected from various smart devices and industrial sensors, each designed to capture specific environmental or operational parameters.

IoT Weather Activity			
ID	**Feature**	**Type**	**Description**
1	date	date	Date of logging IoT telemetry data
2	time	time	Time of logging IoT telemetry data
3	temperature	Number	Temperature measurements of a weather sensor linked to the network
4	pressure	Number	Pressure readings of a weather sensor linked to the network
5	humidity	Number	Humidity readings of a weather sensor linked to the network
6	label	Number	Tag normal and attack records, where 0 indicates normal and 1 indicates attacks
7	type	String	Tag attack categories, such as normal, DoS, DDoS and backdoor attacks, and normal records

Table 1: Attack Type Distribution Across IoT Devices

It includes data from a smart fridge sensor that monitors temperature, a garage door sensor that records door state and transmits signals to mobile devices, a GPS tracker that logs geographical coordinates, and Modbus protocol features widely used in industrial control systems to read registers and coils.

In addition, the dataset incorporates motion sensors to detect movement, light sensors to control lighting, a thermostat sensor that measures and regulates room temperature, and a weather sensor that collects temperature, pressure, and humidity values.

IoT Garage_Door Activity			
ID	**Feature**	**Type**	**Description**
1	date	date	Date of logging IoT telemetry data
2	time	time	Time of logging IoT telemetry data
3	door_state	Boolean	State of a door sensor linked to the network where the door is closed or open
4	sphone_signal	Boolean	State of receiving the door signal on a phone where the signal is true or false
5	label	Number	Tag normal and attack records, where 0 indicates normal and 1 indicates attacks
6	type	String	Tag attack categories, such as normal, DoS, DDoS and backdoor attacks, and normal records

Table 2: IoT Grage Door Features-Description

The features of each IoT device are presented in Tables 1–7, which describe the structure of telemetry data and its relevance to IoT-based intrusion detection. Each record in the dataset is labeled as either normal or attack, with attack types categorized into classes such as DoS, DDoS, and Backdoor. The dataset has a notable class imbalance across attack kinds, as Table 8 illustrates. For example, the Thermostat device captures 15,000 typical events, but only 61 of those are scanning assaults, and 449 are XSS attempts.

Thermostat device has 6 features in its instance vector as shown with feature number in Table 7. Feature number in an instance vector for other devices are presented in Tables 1-6 [Moustafa, 21].

IoT GPS_Tracker Activity			
ID	Feature	Type	Description
1	date	date	Date of logging IoT telemetry data
2	time	time	Time of logging IoT telemetry data
3	latitude	Number	Latitude value of GPS tracker sensor linked to the network
4	longitude	Number	longitude value of GPS tracker sensor linked to the network
5	label	Number	Tag normal and attack records, where 0 indicates normal and 1 indicates attacks
6	type	String	Tag attack categories, such as normal, DoS, DDoS and backdoor attacks, and normal records

Table 3: IoT GPS Tracker Features-Description

IoT Modbus Activity			
ID	Feature	Type	Description
1	date	date	Date of logging IoT telemetry data
2	time	time	Time of logging IoT telemetry data
3	FC1_Read_Input_Register	Number	Modbus function code that is responsible for reading an input register
4	FC2_Read_Discrete_Value	Number	Modbus function code that is in charge of reading a discrete value
5	FC3_Read_Holding_Register	Number	Modbus function code that is responsible for reading a holding register
6	label	Number	Tag normal and attack records, where 0 indicates normal and 1 indicates attacks
7	type	String	Tag attack categories, such as normal, DoS, DDoS and backdoor attacks, and normal records

Table 4: IoT Modbus Features-Description

IoT Fridge Aactivity			
ID	**Feature**	**Type**	**Description**
1	date	date	Date of logging IoT telemetry data
2	time	time	Time of logging IoT telemetry data
3	fridge_temperature	Number	Temperature measurement of a fridge sensor linked to the network
4	temp_condition	String	Temperature conditions of a fridge sensor linked to the network, where temperature is high of low based on a predefined threshold value
5	label	Number	Tag normal and attack records, where 0 indicates normal and 1 indicates attacks
6	type	String	Tag attack categories, such as normal, DoS, DDoS and backdoor attacks, and normal records

Table 5:IoT Fridge Features-Description

IoT Motion_Light Activity			
ID	**Feature**	**Type**	**Description**
1	date	date	Date of logging IoT telemetry data
2	time	time	Time of logging IoT telemetry data
3	motion_status	Number	Status of a motion sensor is either on (i.e., 1) or off (i.e., 0)
4	light_status	Boolean	Status of a light sensor is either on or off
5	label	Number	Tag normal and attack records, where 0 indicates normal and 1 indicates attacks
6	type	String	Tag attack categories, such as normal, DoS, DDoS and backdoor attacks, and normal records

Table 6: IoT Motion Light Features-Description

IoT Thermostat Activity			
ID	**Feature**	**Type**	**Description**
1	date	date	Date of logging IoT telemetry data
2	time	time	Time of logging IoT telemetry data
3	current_temperature	Number	Current Temperature reading of a thermostat sensor connected with the network
4	thermostat_status	Boolean	Status of a thermostat sensor is either on or off
5	label	Number	Tag normal and attack records, where 0 indicates normal and 1 indicates attacks
6	type	String	Tag attack categories, such as normal, DoS, DDoS and backdoor attacks, and normal records

Table 7: IoT Thermostat Features-Description

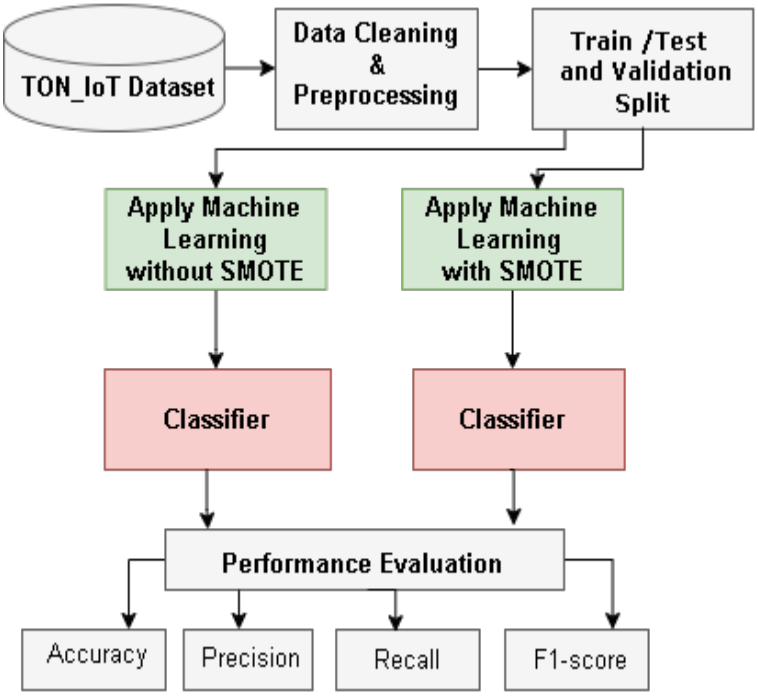

Figure 1: Process of Performance Analysis on the TON_IoT Dataset using SMOTE and Machine Learning

The TON_IoT dataset subsets undergo the following preprocessing pipeline to prepare them for ML models:

The datasets are loaded from source files with comprehensive examination of their dimensions and features. We specifically visualize the class distributions of the target variable to evaluate the effects of imbalance.

Temporal features (date/time) and device status variables, as part of categorical data encoding, are numerically encoded using LabelEncoder, following the approach described by Albasheer [Albasheer et al., 24].

Feature selection is applied. It is observed that two columns can be used as the target class for classification: "type" and "label". The *label* column distinguishes data instances as either 0 for *normal* or 1 for *attack*, whereas the *type* column provides a multi-class description. Because our goal is to perform multi-class classification, we remove the *label* column in the datasets and use the *type* column as the target class for classification.

For the Feature-Target Separation, the data are structured as:

X: Feature matrix (IoT device characteristics) (6-7 features depending on the device)

y: Target vector (attack type classifications)

Stratified Data Partitioning: The dataset is split while preserving class ratios as follows:

70% training

15% validation

15% test

No feature extraction is applied.

Devices	Normal	DDoS	Backdoor	Injection	Password	Ransomware	XSS	Scanning
Fridge	15000	5000	5000	5000	5000	2902	2042	-
Garage Door	15000	5000	5000	5000	5000	2902	1156	529
GPS Tracker	15000	5000	5000	5000	5000	2833	577	550
Modbus	15000	-	5000	5000	5000	-	577	529
Motion Light	15000	5000	5000	5000	5000	2264	449	1775
Weather	15000	5000	5000	5000	5000	2865	866	529
Thermostat	15000	-	5000	5000	5000	2264	449	61

Table 8: Attack Type Distribution Across IoT Devices

3.2 Implementation of Machine Learning Algorithms

This study employs four fundamental ML classifiers —KNN, SVM, LR, and MLP— selected for their proven stability and prevalent use in imbalanced IoT intrusion detection research. KNN is chosen for its flexible, distance-based similarity in small datasets, as described by Cover and Hart [Cover and Hart, 67], while SVM is included for its capacity to construct robust decision boundaries through margin maximization, maintaining efficacy under class imbalance, as established by Cortes and Vapnik [Cortes and Vapnik, 95]. LR provided a reliable probabilistic baseline, as discussed by Mahalaxmi and Rekha [Mahalaxmi and Rekha, 22] and Bhavitha and Thangaraj

[Bhavitha and Thangaraj, 22], and MLP is selected for its ability to model complex, non-linear relationships via backpropagation within a multi-layer architecture, given appropriate preprocessing, as discussed by Taud and Mas [Taud and Mas, 17].

To establish a definitive performance baseline, the experimental phase begins by training and evaluating all four classifiers on the raw, imbalanced IoT device datasets without SMOTE. The data are split into 70% for training, 15% for validation, and 15% for testing. Each model is first tuned and assessed on the validation set, with final generalization performance measured rigorously on the held-out test set. Evaluation is conducted using key metrics derived from the confusion matrix including accuracy, precision, recall, F1-score, and the area under the ROC curve (AUC), as described by Sayegh et al. [Sayegh et al.,24] with particular emphasis on performance disparities across different attack categories. The formal definitions of these evaluation metrics are summarized in Table 9. This initial baseline is critical for cleanly isolating the subsequent impact of the SMOTE balancing technique applied in the next phase of the experiment, as described by Kalifati and Konyar [Kalifati and Konyar, 25].

		Real Values	
		Positive	**Negative**
Estimated Values	**Positive**	True Positive (TP)	False Positive (FP)
	Negative	False Negative (FN)	True Negative (TN)

Table 9: Information on Performance Criteria

The information given in the table is based on the correct or incorrect classification of the data in the classification process. In the light of those, the parameters we used in the paper is explained below.

Accuracy: The ratio of all correct predictions to the total number of predictions.

$$Accuracy = \frac{TN+TP}{TN+TP+FN+FP} \quad (1)$$

Precision: The proportion of predicted positive samples that are truly positive.

$$Precision = \frac{TP}{FP+TP} \quad (2)$$

Recall: The proportion of true positive samples that are correctly detected.

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

F1 Score: The harmonic means of precision and recall; particularly important for imbalanced datasets.

$$F1\ Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (4)$$

AUC (Area Under the Curve): A metric to measure classification success, referring to the area under the ROC curve.

$$AUC = Area\ Under\ the\ ROC\ Curve \quad (5)$$

The hyperparameter settings used for each classifier are summarized in Table 10. These values correspond to the exact configurations applied during model training and validation.

Algorithm	Hyperparameters	Description
KNN	n_neighbors=5; metric=minkowski; p=2	Default scikit-learn Euclidean distance configuration.
SVM	kernel=rbf; C=1.0; gamma=scale; class_weight=balanced; probability=True	RBF kernel with class balancing for imbalanced raw data.
LR	max_iter=1000; solver=lbfgs; multi_class=auto	Increased iteration limit to ensure convergence.
MLP	hidden_layer_sizes=(100,); activation=relu; max_iter=300; learning_rate=constant	One hidden layer with ReLU activation.
SMOTE	k_neighbors=5; sampling_strategy=auto; random_state=42	Default balancing configuration.

Table 10: Hyperparameters Used for Machine Learning Models

3.3 Class Balancing via SMOTE

To address the pronounced class imbalance, we implement the SMOTE with strict adherence to minority-class-only augmentation. The largest class (typically "Normal" traffic) preserves its original sample count, while synthetic samples are strategically generated for under-represented attack categories until achieving numerical parity with the majority class. For example, the Modbus device is our representative case, and attack categories receive standardized encodings as shown in Table 11, enabling systematic evaluation of the balancing process.

Attack Category	Value
Backdoor	0
Injection	1
Normal	2
Password	3
Scanning	4
XSS	5

Table 11: Modbus IoT Attack Categories and Their Numeric Values

This selective oversampling approach maintains dataset integrity while ensuring all classes reach the 15,000-sample threshold of the "Normal" baseline, as visually shown in Figure 2. The same applies to the other six devices.

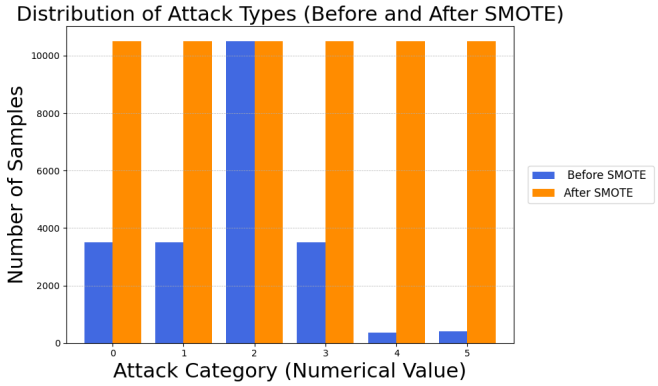

Figure 2: IoT Modbus Category Distribution Before and After SMOTE

4 Performance Evaluation of Attack Detection in IoT Devices

To evaluate the proposed attack-detection framework, we conduct a comprehensive study on seven IoT devices (Fridge, Garage Door, GPS Tracker, Modbus, Motion Light, Thermostat, and Weather) using four standard classifiers —KNN, SVM, LR, and MLP. Each model is trained and tested under two settings: (i) the original imbalanced datasets, which reflect real traffic distributions, and (ii) datasets balanced with the SMOTE. Performance is assessed using accuracy, precision, recall, F1-score, and confusion matrices. While all devices are evaluated using the same protocol and their full results appear in the comparative tables, we provide step-by-step analyses for two representative devices: Garage Door, which shows moderate imbalance with well-separated classes, and Motion Light, which exhibits severe imbalance and sparsity across several attack categories. This contrast illustrates how SMOTE affects learning under different imbalance conditions, whereas the remaining devices follow similar interpretative patterns and are summarized for completeness.

Attack Category	Value
Backdoor	0
DDoS	1
Injection	2
Normal	3
Password	4
Ransomware	5
scanning	6
XSS	7

Table 12: IoT Garage Door Attack Categories and Their Numeric Values

4.1 IoT Garage Door

To provide an overview of the IoT Garage Door dataset and its label structure, Table 12 lists the attack categories and their corresponding numeric values used for model training and evaluation. Figure 3 illustrates the distribution of these categories before and after applying the SMOTE technique, highlighting the initial imbalance and the adjustments achieved through oversampling. To further clarify the extent of this imbalance, Table 13 presents the exact number of samples in each attack category prior to and following the SMOTE process. This detailed comparison supports the interpretation of the model results and helps explain why certain classifiers achieve unusually high performance on classes with very limited original data.

Attack Value	Before SMOTE	After SMOTE
0	3500	10499
1	3500	10499
2	3500	10499
3	10499	10499
4	3500	10499
5	2031	10499
6	370	10499
7	809	10499

Table 13: IoT Garage Door Class Distribution Before and After Applying SMOTE

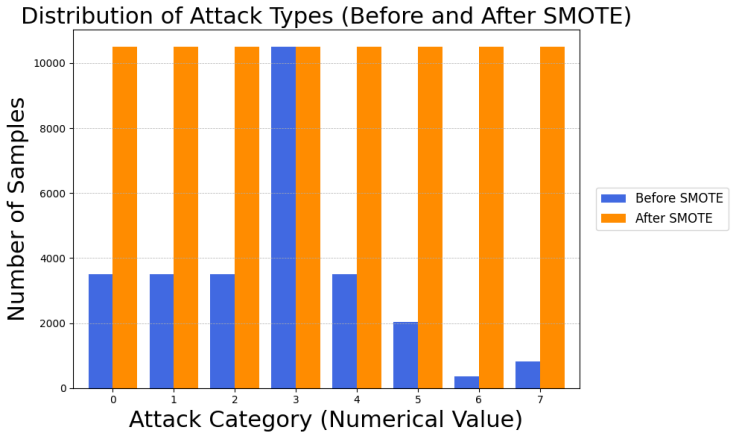

Figure 3: IoT Garage Door Attack Distribution Before and After Applying SMOTE

4.1.1 KNN Algorithm

To evaluate the impact of the SMOTE on the KNN algorithm for the IoT Garage Door dataset, the model is tested under two conditions: using the original imbalanced data and after applying SMOTE. Figure 4 presents the resulting confusion matrices, providing a direct visual comparison of classification performance before and after oversampling. The confusion matrix labels are defined as follows: 0:Backdoor, 1:DDoS, 2:Injection, 3:Normal, 4:Password, 5:Ransomware, 6:Scanning, and 7:XSS. The analysis of the Garage Door device data shows that the Normal (class 3) is the majority class as shown in Figure 3, while several attack types are underrepresented. Before SMOTE, classes such as DDoS (class 1) and XSS (class 7) have significantly fewer samples, resulting in lower classification performance. After applying SMOTE, these minority classes are increased to match the number of samples in the majority class, leading to notable improvements. Correct classifications for DDoS (class 1) increase from 650 to 702, and for XSS (class 7) from 94 to 169. Minor improvements are also observed for Injection (class 2) and Password (class 4), while the majority class Normal (class 3) maintains its high performance without change, as no synthetic samples are generated for it. Overall, SMOTE effectively reduces class imbalance for the Garage Door device, improving minority class accuracy without negatively affecting dominant class performance.

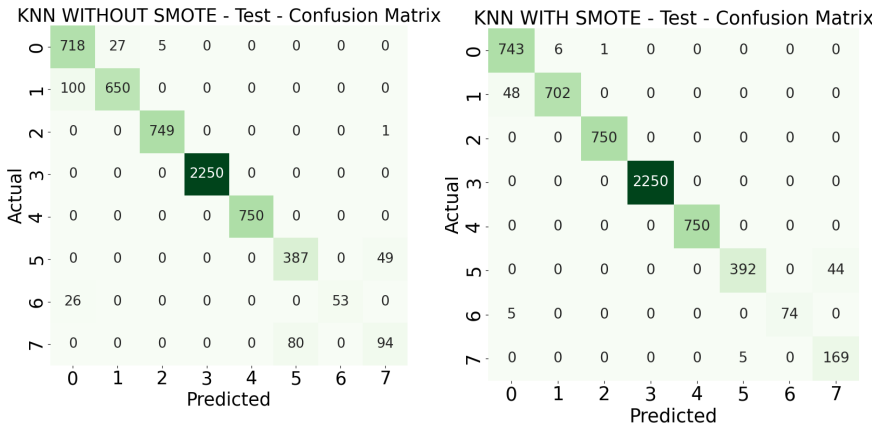

Figure 4: IoT Garage Door KNN with and without SMOTE Confusion Matrix

4.1.2 SVM Algorithm

To assess the impact of the SMOTE on the SVM algorithm for the IoT Garage Door dataset, the model is trained and evaluated under two conditions: with the original imbalanced data and with the balanced dataset produced by SMOTE. Figure 5 presents the results of confusion matrices, enabling a direct visual comparison of the classification performance before and after oversampling. The confusion matrix labels are defined as follows: 0:Backdoor, 1:DDoS, 2:Injection, 3:Normal, 4:Password, 5:Ransomware, 6:Scanning, and 7:XSS.

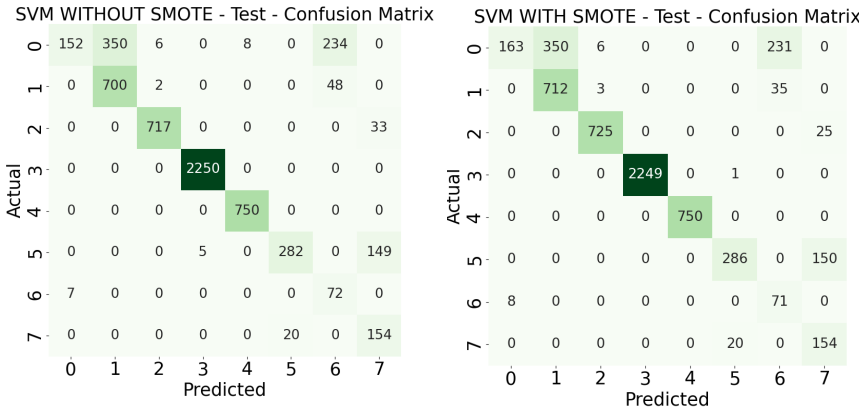

Figure 5: IoT Garage Door SVM with and without SMOTE Confusion Matrix

For the Garage Door device, a comparison of the SVM algorithm's confusion matrices before and after applying SMOTE shows that the most notable improvements occurs in classes with fewer samples. Specifically, correct predictions for the DDoS (class 1) increase from 700 to 712, and for the Injection (class 2) from 717 to 725, indicating that SMOTE provides a positive contribution to underrepresented classes. No changes are observed in classes such as Password (class 4) and XSS (class 7), which already have balanced sample counts, demonstrating that SMOTE does not disrupt the model's existing stability. Minor decreases are recorded in the Normal (class 3) (from 2250 to 2249) and the Scanning (class 6) from 72 to 71, but these are too small to negatively affect overall performance. In this context, SMOTE can be considered a limited yet effective enhancement tool. This is directly related to the SVM algorithm's sensitivity to class boundaries, meaning that the positioning of synthetic samples within the feature space plays a decisive role in SMOTE's effectiveness. Overall, for applications aiming to improve the performance of rare classes, careful and targeted use of SMOTE can be beneficial.

4.1.3 Logistic Regression Algorithm

To evaluate the effect of the SMOTE on the LR algorithm for the IoT Garage Door dataset, the model is tested in two scenarios: using the original imbalanced dataset and after applying SMOTE. Figure 6 presents the resulting confusion matrices, enabling a direct visual comparison of classification performance before and after oversampling. The confusion matrix labels are defined as follows: 0:Backdoor, 1:DDoS, 2:Injection, 3:Normal, 4:Password, 5:Ransomware, 6:Scanning, and 7:XSS.

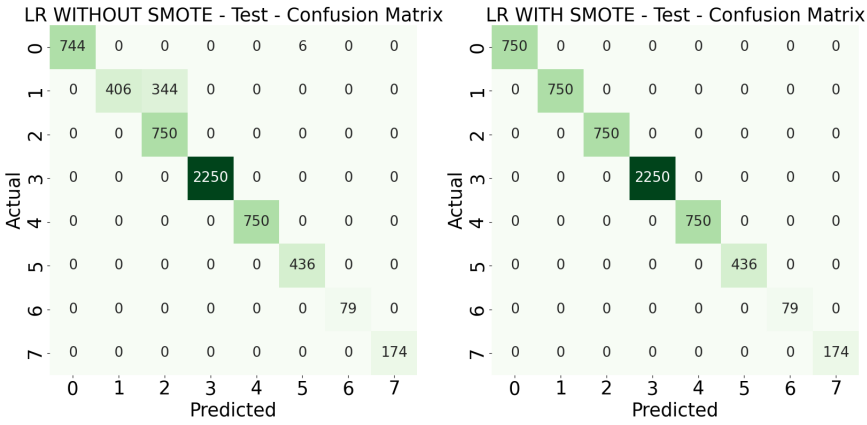

Figure 6: IoT Garage Door LR with and without SMOTE Confusion Matrix

After applying SMOTE, the DDoS (class 1) shows a marked improvement, with correct classifications increasing from 406 to 750. This improvement is attributed to the clearer patterns of this attack type and the more distinct separation between classes, enabling the LR algorithm —sensitive to linear boundaries— to learn this distinction more effectively. In contrast, the Backdoor (class 0) increases only slightly from 744 to 750, reflecting a minimal gain. The results for the other six classes remain entirely unchanged, with no increases or decreases observed. This indicates that SMOTE is not equally effective across all classes; rather, its impact depends directly on the clarity of class boundaries and the distribution of samples. These findings suggest that while SMOTE is not a miracle solution, it does not harm the model and can be beneficial under certain conditions. Overall, the LR algorithm and the Garage Door device show a limited and selective response to SMOTE, with improvements observed only in classes that are inherently easier to classify.

4.1.4 MLP Algorithm

To analyze the effect of balancing on the performance of the MLP algorithm for the IoT Garage Door dataset, the model is trained and evaluated under two conditions: using the original imbalanced dataset and after applying the SMOTE technique. Figure 7 presents the confusion matrices for both cases, allowing a direct comparison of the classification performance before and after oversampling. The confusion matrix labels are defined as follows: 0:Backdoor, 1:DDoS, 2:njection, 3:Normal, 4:Password, 5:Ransomware, 6:Scanning, and 7:XSS.

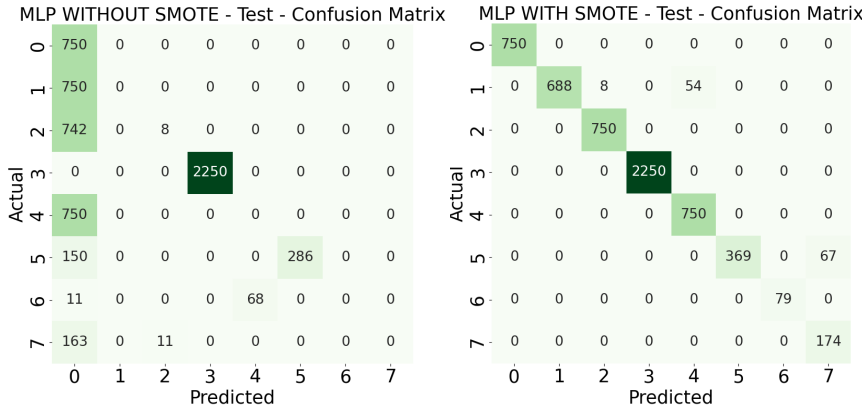

Figure 7: IoT Garage Door MLP with and without SMOTE Confusion Matrix

Before applying SMOTE, the MLP algorithm fails to correctly predict the low-representation classes: DDoS (class 1), Injection (class 2), Password (class 4), Scanning (class 6), and XSS (class 7), each recording zero correct classifications. After SMOTE, the Injection (class 2) increases from 8 to 750 correct predictions. The Password (class 4) and DDoS (class 1) increas from 0 to 750 and 0 to 688, respectively. XSS (class 7) improves from 0 to 174, and Scanning (class 6) from 0 to 79. The Ransomware class (class 5) shows a moderate growth from 286 to 369.

In contrast, the classes with equal instances, such as Backdoor (class 0) and Normal (class 3), remain unchanged at 750 and 2250 correct predictions, indicating that SMOTE does not disrupt their performance. These results show that SMOTE improved only the underrepresented classes while preserving existing balance. The multi-layer, non-linear structure of MLP enables it to learn effectively from the synthetic samples, making SMOTE particularly effective in this case. The pronounced imbalance in the dataset further amplifies SMOTE's contribution. Overall, these findings confirm that SMOTE is a safe and effective balancing method for highly imbalanced, multi-class datasets, especially when used with models oriented towards deep learning, such as MLP.

4.2 IoT Motion Light

To provide an overview of the IoT Motion Light dataset and its label structure, Table 14 lists the attack categories along with their assigned numeric values used during model training and evaluation. Figure 8 illustrates the distribution of these categories before and after applying the SMOTE technique, highlighting the dataset's initial imbalance and the adjustments achieved through oversampling. To further clarify the extent of this imbalance, Table 15 presents the exact number of samples in each attack category prior to and following SMOTE. This detailed comparison supports the discussion on model behavior and explains why some classifiers achieve near-perfect performance on classes with very limited original data.

Attack Category	Value
Backdoor	0
DDoS	1
Injection	2
Normal	3
Password	4
Ransomware	5
scanning	6
XSS	7

Table 14: IoT Motion Light Attack Categories and Their Numeric Values

Attack Value	Before SMOTE	After SMOTE
0	3500	10499
1	3500	10499
2	3500	10499
3	10499	10499
4	3500	10499
5	1584	10499
6	1242	10499
7	314	10499

Table 15: IoT Motion Light Class Distribution Before and After Applying SMOTE

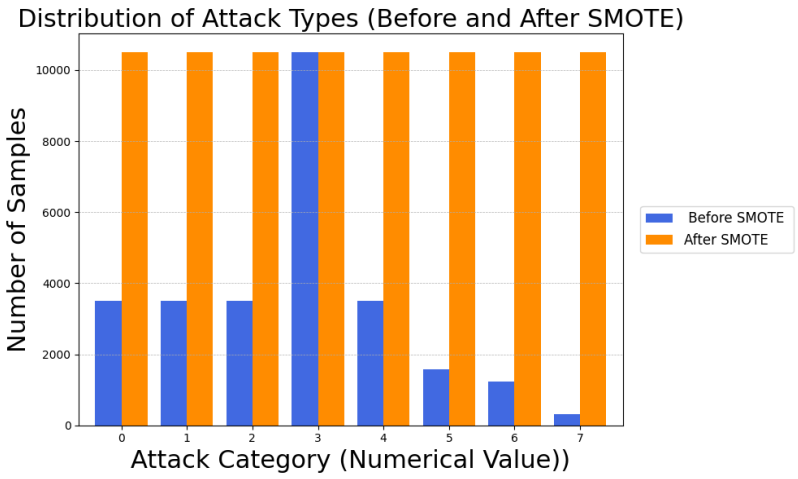

Figure 8: IoT Motion Light Attack Category Distribution Before and After SMOTE

4.2.1 KNN Algorithm

The model is tested using both the original imbalanced data and the balanced dataset produced by SMOTE to evaluate the impact of applying the SMOTE on the

performance of the KNN algorithm for the Motion Light device dataset. Figure 9 displays the corresponding confusion matrices, providing a direct visual comparison of the classification performance in both scenarios. The confusion matrix labels are defined as follows: 0:Backdoor, 1:DDoS, 2:Injection, 3:Normal, 4:Password, 5:Ransomware, 6:Scanning, and 7:XSS.

For the Motion Light device, a comparison of KNN algorithm performance before and after applying SMOTE shows a highly pronounced effect on rare classes. The Scanning (class 6), which initially has only 77 correct classifications, rises to 216 after SMOTE, representing an improvement of approximately 180.5%. Similarly, the XSS (class 7) increases from 24 to 60 correct classifications, marking a 150% improvement. These substantial gains indicate that SMOTE effectively augments underrepresented examples, enabling the model to learn their patterns more successfully. Smaller yet positive changes are observed for classes such as DDoS (class 1) and Password (class 4), where correct classifications increase from 479 to 531 (10.85% gain) and show a 3.65% improvement in recall, respectively. The Injection (class 2) remains unchanged at 750 correct predictions after SMOTE.

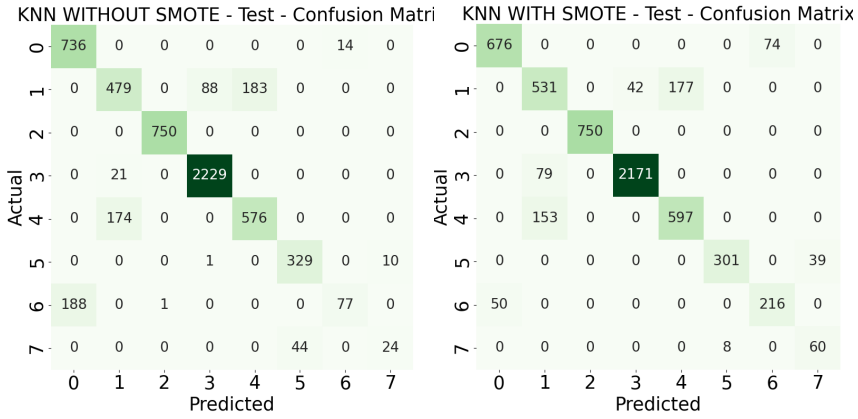

Figure 9: IoT Motion Light KNN with and without SMOTE Confusion Matrix

On the other hand, limited performance drops are noted in large-sample classes. Correct classifications for Normal (class 3) decrease from 2229 to 2171 (-2.6% in recall), for Backdoor (class 0) by -8.15% in recall, and for Ransomware (class 5) by -8.51% in recall. These declines suggest that SMOTE's reshaping of the data distribution may cause some normal instances to be misclassified as attacks. However, given that these decreases are relatively small and high-percentage gains are achieved in rare classes, the overall model performance can be considered positively affected. In conclusion, for the Motion Light device, combining SMOTE with the KNN algorithm successfully reduces class imbalance and significantly improves performance for rare examples. Nevertheless, such effects may not be guaranteed for every device and model, underscoring the need for case-specific evaluation before generalization.

4.2.2 SVM Algorithm

To examine the effect of class balancing on the SVM algorithm when applied to the Motion Light device dataset, the model is evaluated in two configurations: using the original imbalanced data and after applying the SMOTE technique. Figure 10 presents the corresponding confusion matrices, allowing a direct comparison of the algorithm's performance under both conditions. The confusion matrix labels are defined as follows: 0:Backdoor, 1:DDoS, 2:Injection, 3:Normal, 4:Password, 5:Ransomware, 6:Scanning, and 7:XSS.

For the Motion Light device, an analysis of the classification performance of the SVM before and after SMOTE indicates that the overall changes are limited. Although numerical increases are observed in some classes after SMOTE, these gains are mostly small and do not significantly affect the model's overall performance. For example, the DDoS (class 1) improves from 340 to 364 correct classifications, representing a 7.06% increase, while the Password (class 4) records a 2.57% gain (from 700 to 718). Similarly, the Scanning (class 6) improves by 8.82%. However, these increases remain modest and have only a marginal impact on overall accuracy. In contrast, no changes are observed in the Injection (class 2), Normal (class 3), Ransomware (class 5), and XSS (class 7), indicating that SMOTE is ineffective for these categories. Only the Backdoor (class 0) experiences a slight decrease of 2.66% (from 527 to 513), suggesting that SMOTE neither harms model performance nor provides substantial benefits in this case.

Overall, the results indicate that, for the Motion Light device and the specific dataset covering a defined time period, SMOTE's impact is limited. The method does not produce significant improvements, nor does it cause notable performance degradation. This outcome questions the validity of generalizing SMOTE's effectiveness across all scenarios, highlighting the need to test and adapt the method to each device and dataset individually. Notably, analyses on other devices and different temporal samples reveal that SMOTE yields far more substantial improvements, whereas in this specific case, its contribution is marginal.

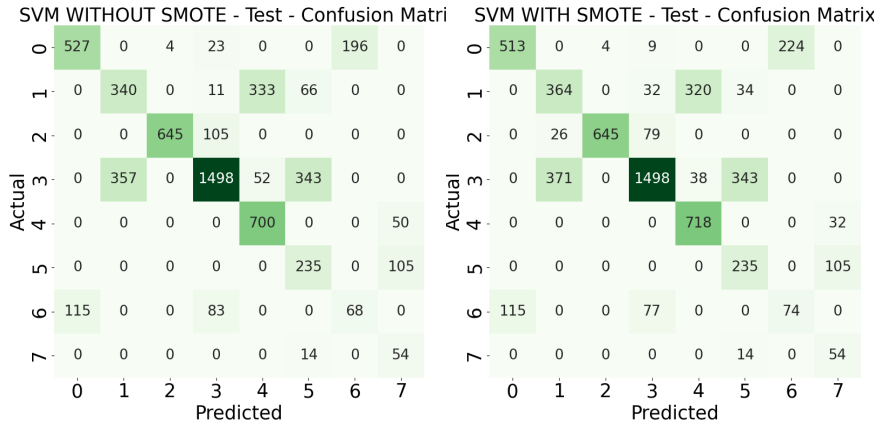

Figure 10: IoT Motion Light SVM with and without SMOTE Confusion Matrix

4.2.3 Logistic Regression Algorithm

To evaluate the effect of the SMOTE on addressing class imbalance in the Motion Light device dataset, LR is applied both before and after SMOTE preprocessing. Figure 11 presents the resulting confusion matrices for both scenarios, enabling a direct comparison of classification performance and the impact of balancing the dataset. The confusion matrix labels are defined as follows: 0:Backdoor, 1:DDoS, 2:Injection, 3:Normal, 4:Password, 5:Ransomware, 6:Scanning, and 7:XSS.

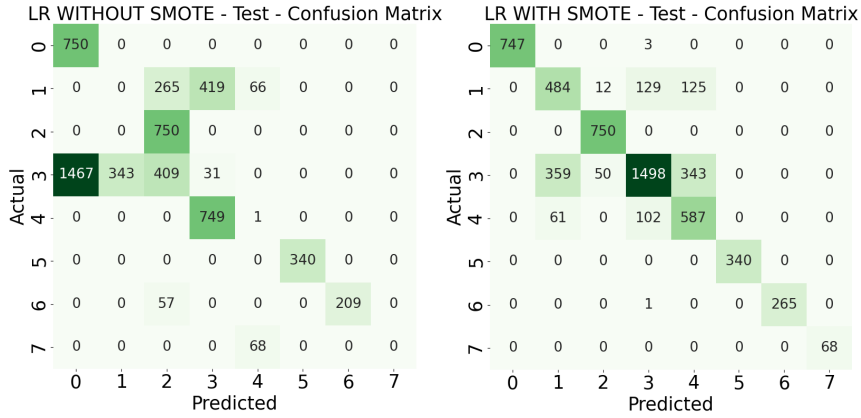

Figure 11: IoT Motion Light LR with and without SMOTE Confusion Matrix

In previous studies, the SVM shows limited effectiveness in addressing class imbalance. However, in the experiment on the Motion Light device dataset, the SMOTE algorithm demonstrates a clear and effective contribution. Notably, the number of samples in class 1 increased from 0 to 484, enabling meaningful representation of a class that is previously entirely overlooked. Similarly, class 3 rises from 31 to 1,498 samples, reflecting an increase of over 4,700%, while class 4 increases from 1 to 587 samples. These results indicate that SMOTE can generate synthetic examples for rare classes in a targeted and controlled manner. The increase in class 6 from 209 to 265 reflects a balanced improvement, and the creation of 68 new samples for class 7, which is initially unrepresented, further demonstrates SMOTE's effectiveness in addressing underrepresented classes. In contrast, sufficiently represented classes such as class 2 (750 samples) and class 5 (340 samples) remain unchanged, showing that the algorithm avoids overproduction and preserves data balance. The slight decrease in class 0 from 750 to 747 is likely due to preprocessing steps rather than a direct effect of SMOTE.

In conclusion, this experiment on the Motion Light dataset demonstrates that SMOTE is an effective tool for correcting class imbalance. The improvement is attributed to both the robust nature of SMOTE and its compatibility with LR. This combination not only balances the dataset but also enhances model performance, highlighting SMOTE and LR as complementary pairs for handling imbalanced data scenarios.

4.2.4 MLP Algorithm

To assess the influence of class balancing on the performance of the MLP algorithm for the Motion Light device dataset, the model is trained and evaluated in two scenarios: before and after applying the SMOTE technique. Figure 12 presents the corresponding confusion matrices, enabling a direct comparison of the algorithm's classification performance under imbalanced and balanced data conditions. The confusion matrix labels are defined as follows: 0:Backdoor, 1:DDoS, 2:Injection, 3:Normal, 4:Password, 5:Ransomware, 6:Scanning, and 7:XSS.

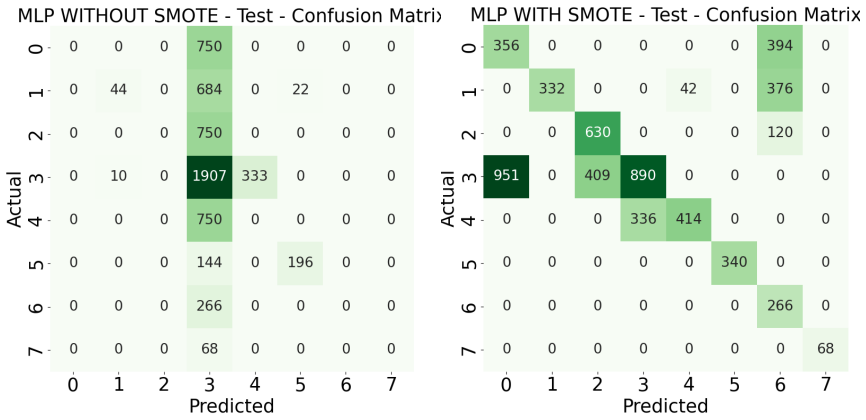

Figure 12: IoT Motion Light MLP with and without SMOTE Confusion Matrix

The results obtained from the Motion Light dataset using the MLP algorithm before and after applying the SMOTE technique, clearly reveal the direct impact of class imbalance on model performance. Initially, the Normal (class 3) is heavily overrepresented with 1,907 samples, while several classes —Backdoor (class 0), Injection (class 2), Password (class 4), Scanning (class 6), and XSS (class 7) —have no samples at all (class 0). This structural imbalance causes the model to focus predominantly on the majority class, effectively ignoring rare attack types. This highlights a fundamental weakness of neural network–based models such as MLP when working with imbalanced datasets. Following the application of SMOTE, synthetic samples are generated and integrated into the dataset for the previously underrepresented classes: 356 for Backdoor (class 0), 630 for Injection (class 2), 414 for Password (class 4), 266 for Scanning (class 6), and 68 for XSS (class 7). Significant increases are also observed in the underrepresented DDoS (class 1) and Ransomware (class 5) classes, rising from 44 to 332 and from 196 to 340, respectively. This rebalancing process expands the learning capacity of the MLP algorithm beyond the dominant class, resulting in a notable improvement in the model's discriminative ability.

However, these results must be interpreted considering the MLP algorithm's inherent characteristics. MLPs are highly sensitive to data distribution and lack built-in mechanisms to compensate for class imbalance. Therefore, the observed improvements reflect not only the effectiveness of SMOTE but also the limitations of MLP in learning

from rare classes. It should also be noted that the synthetic samples produced by SMOTE may not perfectly capture the true data distribution. In sensitive domains such as cybersecurity, careful consideration is required to assess how well these artificially generated samples represent real-world system behaviors. Consequently, the improvements observed should be interpreted contextually and not reduced to numerical gains alone.

In conclusion, while the MLP algorithm can achieve significant performance gains in a balanced setting, it exhibits substantial limitations in the original imbalanced scenario. This demonstrates that SMOTE serves not only as a balancing tool but also as an analytical instrument for exposing model vulnerabilities. Future work should consider using more robust models or enhancing MLP with techniques such as weighting and resampling to better address class imbalance.

5 Results for Comparative Performance Analysis

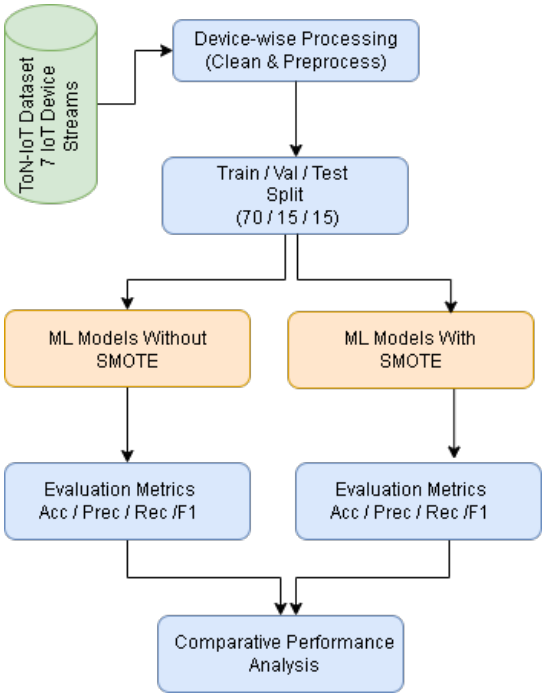

Figure 13: Experimental Workflow

To clarify the structure of the experimental design, Figure 13 presents the complete workflow applied in Section 5. The pipeline begins with device-level preprocessing of the ToN_IoT streams, followed by a stratified 70/15/15 split into training, validation, and test subsets. KNN, SVM, LR, and MLP are trained under two configurations,

original imbalanced data and SMOTE-balanced data, and subsequently evaluated using accuracy, precision, recall, and F1-score. This workflow enables a rigorous comparison of performance before and after SMOTE across all seven IoT devices.

The performance of KNN, SVM, LR, and MLP is evaluated before and after the SMOTE application using accuracy, precision, recall, and F1-score. Tables 16–19 present the detailed results for each algorithm across different IoT devices.

5.1 KNN Performance

As shown in Table 16, The application of the SMOTE generally improves the performance of the KNN algorithm across the majority of IoT devices. The most notable enhancement is observed for the Garage Door device, where accuracy increases from 0.952 to 0.982 and recall from 0.87 to 0.97, indicating that SMOTE significantly enhances the model's ability to correctly identify minority classes.

Substantial improvements are also recorded for the Modbus dataset, with accuracy rising from 0.752 to 0.848 and recall showing a remarkable increase from 0.52 to 0.84, suggesting that the initial performance limitations are largely due to severe class imbalance.

Similarly, Motion Light exhibited an accuracy improvement from 0.878 to 0.895 and a recall increase from 0.75 to 0.87, reflecting SMOTE's positive impact on datasets with pronounced imbalance.

IoT Device	SMOTE	Accuracy	Precision	Recall	F1-score
Fridge	No	0.947	0.90	0.90	0.90
	Yes	0.955	0.91	0.91	0.91
Garage Door	No	0.952	0.91	0.87	0.88
	Yes	0.982	0.96	0.97	0.96
GPS-Tracker	No	0.993	1.00	0.99	1.00
	Yes	0.985	0.99	0.99	0.99
Modbus	No	0.752	0.65	0.52	0.54
	Yes	0.848	0.75	0.84	0.78
Motion Light	No	0.878	0.83	0.75	0.76
	Yes	0.895	0.84	0.87	0.85
Thermostat	No	0.983	0.80	0.77	0.78
	Yes	0.971	0.80	0.90	0.81
Weather	No	0.996	1.00	0.99	0.99
	Yes	0.999	1.00	1.00	1.00

Table 16: KNN Test Set Performance (Before and After SMOTE)

For devices such as Fridge and Weather, which already demonstrate high performance prior to oversampling, improvements are marginal (e.g., recall from 0.90 to 0.91). This suggests that these datasets are relatively balanced or inherently well-suited for KNN classification, reducing the necessity for synthetic sample generation.

Conversely, slight performance declines are noted for Thermostat (accuracy from 0.983 to 0.971) and GPS Tracker (accuracy from 0.993 to 0.985). These reductions may be attributed to the introduction of synthetic samples that, while benefiting minority classes, slightly blurred class boundaries for majority classes. Overall, SMOTE proves

to be most beneficial for devices with highly imbalanced class distributions, substantially improving recall and F1-scores for minority classes. However, its application to already well-balanced datasets should be carefully evaluated, as it may yield negligible benefits or cause minor performance degradation.

5.2 SVM Performance

In Table 17, The application of the SMOTE generally resulted in modest performance gains for the SVM algorithm across most IoT devices, with the extent of improvement varying by dataset. The most substantial enhancement is observed for the Thermostat device, where accuracy increases from 0.726 to 0.846, accompanied by improvements in recall (0.81 to 0.85) and F1-score (0.62 to 0.70). This suggests that SMOTE is effective in addressing class imbalance for this dataset, enabling SVM to better capture decision boundaries for minority classes.

IoT Device	SMOTE	Accuracy	Precision	Recall	F1-score
Fridge	No	0.911	0.86	0.88	0.86
	Yes	0.922	0.88	0.89	0.87
Garage Door	No	0.855	0.77	0.82	0.72
	Yes	0.860	0.78	0.82	0.73
GPS Tracker	No	0.777	0.73	0.82	0.73
	Yes	0.786	0.74	0.83	0.75
Modbus	No	0.740	0.67	0.75	0.64
	Yes	0.786	0.70	0.81	0.69
Motion Light	No	0.687	0.59	0.67	0.60
	Yes	0.692	0.59	0.68	0.61
Thermostat	No	0.726	0.65	0.81	0.62
	Yes	0.846	0.72	0.85	0.70
Weather	No	0.944	0.87	0.94	0.89
	Yes	0.954	0.89	0.95	0.91

Table17: SVM Test Set Performance (Before and After SMOTE)

Moderate improvements are also recorded for Modbus (accuracy: 0.740 to 0.786, recall: 0.75 to 0.81, F1-score: 0.64 to 0.69) and the Fridge (accuracy: 0.911 to 0.922, recall: 0.88 to 0.89), indicating a positive impact of oversampling on datasets with moderate imbalance. Weather also demonstrates slight yet consistent improvements across all metrics, reflecting SMOTE's stability when applied to already well-performing datasets. In contrast, performance changes for Garage Door, GPS Tracker, and Motion Light are minimal, with accuracy increases of only 0.005, 0.009, and 0.005, respectively. These small variations suggest that SMOTE has a limited influence in scenarios where class separation is already well-defined or where the algorithm's kernel-based nature is less sensitive to synthetic data generation.

Overall, SMOTE's integration with SVM yields the most pronounced benefits for datasets with moderate to high imbalance, particularly those where minority classes can be effectively represented in the feature space. However, for devices with either inherently well-separated classes or near-balanced distributions, the effect remains limited, underscoring the importance of dataset-specific evaluation before applying oversampling techniques.

5.3 Logistic Regression Performance

As shown in Table 18, The application of the SMOTE yields substantial performance improvements for the LR across all IoT devices, with particularly striking gains for datasets exhibiting severe class imbalance. The most remarkable cases are observed in Fridge, Garage Door, GPS Tracker, and Modbus, where accuracy, precision, recall, and F1-score all increase to a perfect 1.000 after SMOTE application.

These results indicate that oversampling successfully eliminates misclassifications by providing balanced and well-represented class distributions, allowing LR's linear decision boundaries to generalize more effectively. For the Motion Light dataset, accuracy rises dramatically from 0.351 to 0.800, with corresponding increases in recall (0.48 to 0.89) and F1-score (0.38 to 0.87). This substantial gain underscores the algorithm's sensitivity to extreme imbalance and the capacity of SMOTE to correct such disparities. Similarly, Thermostat experiences a notable boost in recall (0.61 to 0.93) and accuracy (0.703 to 0.799), reflecting improved recognition of minority classes without sacrificing performance on majority classes. The Weather dataset also demonstrates significant improvement, with accuracy increasing from 0.921 to 0.984 and F1-score from 0.69 to 0.98, confirming that even high-performing datasets can benefit from targeted oversampling when class imbalance exists.

Overall, SMOTE's integration with LR proves to be highly effective across diverse IoT datasets, particularly for scenarios characterized by severe imbalance. The results suggest that the combination of LR's interpretability and SMOTE's balancing capability can deliver both accuracy and robustness, making it a compelling approach for multi-class classification in imbalanced data environments.

IoT Device	SMOTE	Accuracy	Precision	Recall	F1-score
Fridge	No	0.857	0.72	0.72	0.71
	Yes	0.997	1.00	1.00	1.00
Garage Door	No	0.941	0.96	0.94	0.94
	Yes	1.000	1.00	1.00	1.00
GPS Tracker	No	0.767	0.82	0.61	0.61
	Yes	0.997	1.00	1.00	1.00
Modbus	No	0.786	0.52	0.48	0.49
	Yes	1.000	1.00	1.00	1.00
Motion Light	No	0.351	0.36	0.48	0.38
	Yes	0.800	0.86	0.89	0.87
Thermostat	No	0.703	0.67	0.61	0.62
	Yes	0.799	0.79	0.93	0.78
Weather	No	0.921	0.68	0.70	0.69
	Yes	0.984	0.99	0.98	0.98

Table 18: Logistic Regression Test Set Performance (Before and After SMOTE)

5.4 MLP Algorithm

According to the results in Table 19, The application of the SMOTE leads to substantial improvements in the performance of the MLP across nearly all IoT devices, with particularly significant gains for datasets characterized by severe class imbalance. The most notable increase is observed in the Garage Door dataset, where accuracy rises sharply from 0.555 to 0.978, and F1-score improves from 0.27 to 0.96. This dramatic

enhancement reflects MLP's ability to effectively learn from the enriched representation of minority classes created by SMOTE.

IoT Device	SMOTE	Accuracy	Precision	Recall	F1-score
Fridge	No	0.695	0.67	0.57	0.50
	Yes	0.834	0.72	0.80	0.75
Garage Door	No	0.555	0.33	0.33	0.27
	Yes	0.978	0.96	0.97	0.96
GPS Tracker	No	0.654	0.50	0.42	0.40
	Yes	0.921	0.93	0.91	0.92
Modbus	No	0.731	0.70	0.54	0.55
	Yes	0.839	0.76	0.84	0.74
Motion Light	No	0.362	0.26	0.19	0.16
	Yes	0.556	0.72	0.71	0.65
Thermostat	No	0.665	0.47	0.41	0.41
	Yes	0.730	0.72	0.67	0.65
Weather	No	0.870	0.78	0.80	0.78
	Yes	0.959	0.97	0.94	0.95

Table 19: MLP Test Set Performance (Before and After SMOTE)

Similarly, the GPS Tracker dataset exhibites a marked improvement, with accuracy increasing from 0.654 to 0.921 and F1-score from 0.40 to 0.92. Large gains are also observed for Fridge (accuracy: 0.695 to 0.834, recall: 0.57 to 0.80) and Weather (accuracy: 0.870 to 0.959, F1-score: 0.78 to 0.95), indicating that oversampling not only improves minority class recognition but also enhances overall model generalization.

Moderate but meaningful improvements are achieved for Modbus (accuracy: 0.731 to 0.839, recall: 0.54 to 0.84) and Thermostat (accuracy: 0.665 to 0.730, recall: 0.41 to 0.67), demonstrating SMOTE's effectiveness even in datasets with less extreme imbalance. The Motion Light dataset, while showing the lowest post-SMOTE accuracy among all devices (0.556), still experiences a notable jump from its pre-SMOTE value of 0.362, with recall improving from 0.19 to 0.71.

Overall, SMOTE significantly enhances the MLP algorithm's capacity to handle imbalanced data by providing balanced training samples that allow the model to learn minority class patterns more effectively. These results highlight that deep learning–oriented models such as MLP can benefit greatly from oversampling strategies, particularly when faced with skewed multi-class distributions.

5.5 Overall Performance Analysis

Across all seven IoT devices, the impact of SMOTE on classifier performance shows clear and consistent patterns. LR and MLP achieve the most substantial improvements, especially in highly imbalanced devices such as Garage Door, Modbus, and GPS Tracker, where recall and F1-scores increase dramatically after balancing. KNN demonstratsd moderate yet stable gains, particularly for Modbus and Garage Door, reflecting its sensitivity to improved minority-class representation.

For SVM, the improvements are more device dependent. While several devices — such as Fridge, Garage Door, Motion Light, and Weather— show only modest accuracy increases (typically between +0.005 and +0.011), two devices exhibit noticeably larger

gains. Modbus improves by +0.046, and Thermostat shows a substantial increase of +0.120 in accuracy, indicating that SVM benefits more significantly when minority classes are extremely underrepresented or when oversampling enhances feature separability.

Overall, the results indicate that SMOTE is most effective in scenarios where minority classes contain very limited samples and class boundaries are difficult to learn. Algorithms with linear or neural decision boundaries (LR and MLP) benefit the most, while distance-based and kernel-based models (KNN and SVM) show improvements that vary according to the degree of imbalance and device-specific feature characteristics. These findings emphasize the importance of aligning oversampling strategies with both the dataset structure and the classifier's learning behavior.

6 Discussion

The experimental results across seven IoT devices clearly show that class imbalance directly affects model performance, with SMOTE providing substantial improvements especially in devices where minority classes are severely underrepresented. This issue is evident in Garage Door and Motion Light, where several attack categories contain only a few hundred samples, leading to weak baseline performance in the confusion matrices. After applying SMOTE, minority classes are expanded to match the majority class, enabling classifiers—particularly LR and MLP—to learn more consistent decision boundaries. For example, MLP's accuracy on the Garage Door device rises from 0.555 to 0.978, while LR reaches perfect accuracy (1.000) on devices such as Fridge, Garage Door, and Modbus. KNN shows moderate but stable gains (e.g., from 0.752 to 0.848 on Modbus), whereas SVM exhibits selective improvements, with the most notable increase on Thermostat (0.726 to 0.846).

Some devices display slight performance reductions after balancing, such as KNN dropping from 0.993 to 0.985 for GPS Tracker and from 0.983 to 0.971 for Thermostat. These minor declines are expected because synthetic samples can slightly alter class boundaries, though they do not indicate model instability or overfitting. In devices where the original distribution is already balanced—such as Fridge and Weather—SMOTE leads to only marginal changes, confirming that it does not artificially inflate performance when imbalance is minimal.

Overall, SMOTE is highly effective for improving minority-class detection when classes are sparsely represented, but its impact depends on the dataset's distribution and the classifier's sensitivity to sample density, indicating that oversampling should be applied selectively.

Table 20 provides a comparative summary of recent studies that have addressed class imbalance in IoT intrusion detection. The table highlights distinct methodological approaches and their respective limitations, against which the contributions of this study can be clearly positioned.

Ref.	Dataset	Feature Preprocessing	Classifier	Class	Key Result	Limitation
Tasnim et al., 2022	ToN_IoT (Fridge)	Basic cleaning, SMOTE (single device)	RF	8	99% acc	Balancing on one device only (Fridge)
Calp & Bütüner, 2024	NSL-KDD	Standard preprocessing	RF	2	99.6% acc	No multi-device analysis
Gad et al., 2021	TON_IoT (Network dataset)	One-Hot Encoding Min-Max Normalization	XGBoost	2	99.1% acc	Focused on vehicles only
Alotaibi & Ilyas, 2023	TON_IoT	Standard Preprocessing + Feature Selection + SMOTE	DT, ET, RF, KNN, XGB + Voting Ensemble	7	98.9% acc	High complexity
Popoola et al., 2021	Bot-IoT	SMOTE-DRNN	DRNN	11	100% acc 99.5% F1	High computational complexity
Cao et al., 2025	TON_IoT	SMOTE + Tomek Link	CNN	10	98.87 % acc	Data processing complexity
Venkatasubramanian et al., 2025	BoT-IoT, ToN-IoT, CIC-DDoS	SMOTE + BKA	BERT, RoBERTa, XLNet	>20	96.1% acc	Transformer cost
Eren & Küçük, 2023	TON-IoT telemetry	Extensive feature engineering + Automated feature generation + SMOTE/SCUT	RF	10	99.99 % AUC-ROC	Low precision for MITM
This Study	TON-IoT (Garage Door)	Device-level preprocessing + SMOTE	MLP	8	97.8% acc	Few samples in minority classes
This Study	ToN-IoT (GPS Tracker)	Device-level preprocessing + SMOTE	LR	8	99.7% acc	Few samples in minority classes

Table 20: Comparative Summary of Recent IoT Intrusion Detection Studies

Previous research on IoT intrusion detection has demonstrated significant progress in addressing data imbalance and enhancing model performance; however, most studies

remain limited in scope or methodology. For instance, Tasnim et al. [Tasnim et al., 23] achieved over 99% accuracy with RF using SMOTE on the Fridge of the ToN_IoT dataset. Their analysis is confined to a single device, which limits its generalizability. Calp and Bütüner [Calp and Bütüner, 24] attain an impressive accuracy of 99.6% with the RF algorithm on the NSL-KDD dataset, but their work is restricted to binary classification and a single dataset type. Similarly, Gad et al. [Gad et al., 21] achieved an accuracy of 99.1% using the XGBoost algorithm on the ToN_IoT dataset with One-Hot Encoding and Min-Max Normalization. Their study is also limited to binary classification within a single network environment.

Alotaibi and Ilyas [Alotaibi and Ilyas, 23] report an accuracy of 98.9% on the ToN_IoT dataset using an ensemble of DT, RF, KNN, and XGBoost with SMOTE and feature selection. The limitation of their model is that it entails high computational complexity due to the ensemble design. Although Popoola et al. [Popoola et al., 21] achieve nearly perfect accuracy (approximately 100%) and an F1-score of 99.5% using a DRNN with SMOTE on the BoT-IoT dataset, their method suffers from high computational demands and limited generalizability. Cao et al. [Cao et al., 24] obtain an accuracy of 98.87% on the ToN_IoT dataset using a CNN model combined with SMOTE and Tomek Links. However, their approach requires extensive preprocessing and incurres high computational complexity.

Venkatasubramanian et al. [Venkatasubramanian et al., 25] propose a high-performance model using an ensemble of Transformers (BERT, RoBERTa, XLNet), enhanced by SMOTE and the Black-winged Kite Algorithm (BKA) for feature selection, achieving an accuracy of 96.1% on a combined dataset of BoT-IoT, ToN_IoT, and CIC-DDoS2019 with over 20 attack classes. Nevertheless, this approach relies on computationally expensive architectures and complex optimization. While Eren and Küçük [Eren and Küçük, 23] demonstrate the efficacy of automated feature engineering combined with RF on the ToN_IoT dataset, achieving 99.99% AUC-ROC, they also obtain a low precision for MITM.

In contrast to these studies, the difference from previous studies is that our work differentiates itself by adopting a multi-class classification framework across heterogeneous IoT data sources, performing device-level analyses on seven IoT devices, and evaluating the impact of SMOTE on class imbalance. Our approach provides a more realistic representation of real-world intrusion scenarios and a more detailed, device-specific understanding of intrusion behavior. Furthermore, our study achieves competitive performance using simpler, classical models enhanced by SMOTE, thereby offering a more computationally efficient, scalable, and interpretable alternative for diverse IoT environments.

7 Conclusion

This study sets out to address the challenge of class imbalance in IoT intrusion detection and to evaluate whether SMOTE can effectively improve the recognition of ork performs a multi-device minority attack class across multiple IoT devices. This objective is successfully achieved through a device-level evaluation of seven ToN_IoT datasets. The results demonstrate that SMOTE noticeably enhances recall and F1-score for highly imbalanced devices, particularly when paired with LR and MLP, while generating moderate improvements for KNN and SVM. These findings provide

practical insight into how imbalance severity and algorithmic characteristics jointly influence intrusion-detection performance.

The originality of this study arises from its focus on device-specific analysis across seven heterogeneous IoT environments, combined with a systematic comparison of four classical ML algorithms –KNN, SVM, LR, and MLP– under both imbalanced and SMOTE-balanced conditions. Instead of treating the ToN_IoT dataset as a single unified source, the study evaluates each device independently, allowing a clearer understanding of how class imbalance and oversampling techniques influence model behavior in different operational contexts.

This work contributes to the existing literature by providing a detailed, side-by-side examination of performance changes before and after applying SMOTE, highlighting how improvements vary across devices and attack categories. Through this approach, the study offers practical insights into when oversampling is beneficial, how different classifiers react to balanced data, and which IoT environments are most sensitive to class imbalance. These findings enrich the understanding of data-level techniques for intrusion detection and support the development of more robust preprocessing pipelines for IoT security systems.

Despite these contributions, several limitations remain. The effectiveness of SMOTE is influenced by the structure of each device's telemetry data, and the method does not fully resolve imbalance for classes with extremely limited samples. This further contributes the inherent tendency of certain models to overfit such as XGBoost, deep learning architectures. This phenomenon is explicitly reported in the study by Lai et al. [Lai et al., 2021]. Each of their three methods shows overfitting although they perform better on the training set than on the test set. Importantly, this is observed even on datasets that are not severely imbalanced, indicating that the issue relates to model complexity rather than class ratios alone. Additionally, our preliminary experiment with XGBoost on the Motion Light device confirmed this pattern, producing near-perfect accuracy that does not generalize well. To avoid misleading interpretations in the remaining devices, we opt to rely on more stable baseline algorithms that are less prone to overfitting and easier to compare.

Future work is planned to expand on this foundation by evaluating additional resampling techniques such as Borderline-SMOTE, SMOTE-ENN, and ADASYN, integrating cost-sensitive learning, ensemble methods, and exploring deep learning architectures for more robust performance. Testing these approaches on larger and more diverse IoT datasets, including real-time streaming environments, is considered to further strengthen the applicability of intrusion detection systems in practical IoT deployments.

Acknowledgments

This study is supported by the Scientific and Technological Research Council of Turkey (TUBITAK) under Grant Number 124E307. The authors thank TUBITAK for their support. This work has been supported by Kocaeli University Scientific Research Projects Coordination Unit under grant number FAA-2024-4232.

References

[Adi et al., 20] Adi, E., Anwar, A., Baig, Z., Zeadally, S.: "Machine learning and data analytics for the IoT." Neural Comput. Appl., 32(20), 2020, 16205–16233.

[Ait Mouha, 21] Ait Mouha, R. A. R.: "Internet of Things (IoT)." J. Data Anal. Inf. Process., 9(2), 2021, 77–101.

[Al-Areqi and Konyar, 22] Al-Areqi, F., & Konyar, M. Z.. "Effectiveness evaluation of different feature extraction methods for classification of covid-19 from computed tomography images: A high accuracy classification study." Biomedical Signal Processing and Control, 76, 2022, 103662.

[Albasheer et al., 24] Albasheer, F. O., Haibatti, R. R., Agarwal, M., Nam, S. Y.: "A novel IDS based on Jaya optimizer and SMOTE-ENN for cyberattacks detection." IEEE Access, 12, 2024, 101506–101527.

[Alotaibi and Ilyas, 23] Alotaibi, Y., Ilyas, M.: "Enhancing IoT attack detection through ensemble-based multiclass attacks classification." In Proc. 2023 IEEE 20th International Conference on Smart Communities: Improving Quality of Life using AI, Robotics and IoT (HONET), 2023, 1–6. IEEE.

[Alsaedi et al., 20] Alsaedi, A., Moustafa, N., Tari, Z., Mahmood, A., Anwar, A.: "TON_IoT telemetry dataset: A new generation dataset of IoT and IIoT for data-driven intrusion detection systems." IEEE Access, 8, 2020, 165130–165150.

[Bhavitha and Thangaraj, 22] Bhavitha, K. V., Thangaraj, S. J. J.: "Novel detection of accurate spam content using logistic regression algorithm compared with Gaussian algorithm." In Proc. 2022 International Conference on Business Analytics for Technology and Security (ICBATS), 2022, 1–5. IEEE.

[Blagus and Lusa, 13] Blagus, R., Lusa, L.: "SMOTE for high-dimensional class-imbalanced data." BMC Bioinformatics, 14(1), 2013, 106.

[Calp and Bütüner, 24] Calp, M. H., Bütüner, R.: "Detecting the cyber attacks on IoT-based network devices using machine learning algorithms." Politeknik Dergisi, 27(5), 2024, 1971–1989.

[Cao et al., 25] Cao, Z., Zhao, Z., Shang, W., Ai, S., Shen, S.: "Using the ToN-IoT dataset to develop a new intrusion detection system for industrial IoT devices." Multimedia Tools and Applications, 84(16), 2025, 16425–16453.

[Cortes and Vapnik, 95]Cortes, C., Vapnik, V.:"Support-vector networks." Machine Learning, 20(3), 1995, 273–297.

[Cover and Hart, 67]Cover, T., Hart, P.:"Nearest neighbor pattern classification." IEEE Transactions on Information Theory, 13(1), 1967, 21–27.

[Dablain et al., 22] Dablain, D., Krawczyk, B., Chawla, N. V.: "DeepSMOTE: Fusing deep learning and SMOTE for imbalanced data." IEEE Trans. Neural Netw. Learn. Syst., 34(9), 2022, 6390–6404.

[Doghramachi and Ameen, 23] Doghramachi, D. F., Ameen, S. Y.: "Internet of Things (IoT) security enhancement using XGBoost machine learning techniques." Comput. Mater. Contin., 77(1), 2023, 717–732.

[Eid et al., 24] Eid, A. M., Soudan, B., Nassif, A. B., Injadat, M.: "Comparative study of ML models for IIoT intrusion detection: impact of data preprocessing and balancing." Neural Comput. Appl., 36(13), 2024, 6955–6972.

[Eren and Küçük, 23] Eren, K. K., Küçük, K.: "Improving intrusion detection systems for IoT devices using automated feature generation based on ToN_IoT dataset." In Proc. 8th Int. Conf. Computer Science and Engineering (UBMK), 2023, 276–281. IEEE.

[Farea et al., 24] Farea, A. H., Alhazmi, O. H., Kucuk, K.: "Advanced optimized anomaly detection system for IoT cyberattacks using artificial intelligence." Comput. Mater. Contin., 78(2), 2024, 1525–1545.

[Gad et al., 21] Gad, A. R., Nashat, A. A., Barkat, T. M.: "Intrusion detection system using machine learning for vehicular ad hoc networks based on ToN-IoT dataset." IEEE Access, 9, 2021, 142206–142217.

[Gokhale et al., 18] Gokhale, P., Bhat, O., Bhat, S.: "Introduction to IOT." Int. Adv. Res. J. Sci. Eng. Tech., 5(1), 2018, 41–44.

[Hussain et al., 20] Hussain, F., Hussain, R., Hassan, S. A., Hossain, E.: "Machine learning in IoT security: Current solutions and future challenges." IEEE Commun. Surveys Tutorials, 22(3), 2020, 1686–1721.

[Kalifati et al., 25] Kalifati, E., Küçük, K., Şolpan, Ş., Eren, K. K.: "A Comparative Study of SMOTE-Enhanced Machine Learning Models for IoT Device Classification on the TON_IoT Dataset." In Proc. 2025 IEEE International Black Sea Conference on Communications and Networking (BlackSeaCom), 2025, 1–6. IEEE.

[Kalifati and Konyar, 25] Kalifati, S., Konyar, M. Z. “Brain Tumor Detection Using Explainable Artificial Intelligence and Transformers”. In 2025 33rd Signal Processing and Communications Applications Conference (SIU), 2025, June, pp. 1-4. IEEE.

[Kocaoğlu et al., 22] Kocaoğlu, D., Turgut, K., and Konyar, M. Z. "Sector-based stock price prediction with machine learning models." Sakarya University Journal of Computer and Information Sciences, *5*(3), 2022, 415-426.

[Lai et al., 21] Lai, S.B. S., Shahri, N. H. N. B. M., Mohamad, M. B., Rahman, H. A. B. A., Rambli, A. B.: "Comparing the performance of AdaBoost, XGBoost, and logistic regression for imbalanced data." Mathematics and Statistics, 9(3), 2021, 379–385.

[Mahalaxmi and Rekha, 22] Mahalaxmi, K. V. K., Rekha, K. S.: "Comparison of logistic regression and artificial neural network for modelling credit card data set with the identification of precise fraudulent." In Proc. 2022 International Conference on Business Analytics for Technology and Security (ICBATS), 2022, 1–5. IEEE.

[Mazhar et al., 23] Mazhar, T., Talpur, D. B., Shloul, T. A., Ghadi, Y. Y., Haq, I., Ullah, I., Ouahada, K., Hamam, H.: "Analysis of IoT security challenges and its solutions using artificial intelligence." Brain Sci., 13(4), 2023, 683.

[Mohammed et al., 20] Mohammed, A. J., Hassan, M. M., Kadir, D. H.: "Improving classification performance for a novel imbalanced medical dataset using SMOTE method." Int. J. Adv. Trends Comput. Sci. Eng., 9(3), 2020, 3161–3172.

[Mohindru et al., 21] Mohindru, G., Mondal, K., Banka, H.: "Different hybrid machine intelligence techniques for handling IoT-based imbalanced data." CAAI Trans. Intell. Technol., 6(4), 2021, 405–416.

[Moustafa, 21] Moustafa, N.: "A new distributed architecture for evaluating AI-based security systems at the edge: Network TON_IoT datasets." Sustain. Cities Soc., 72, 2021, 102994.

[Öngir et al., 25] Öngir, S., Kaleli, E. C., Konyar, M. Z., and Ertunç, H. M. "Vertical Force Monitoring of Racing Tires: A Novel Deep Neural Network-Based Estimation Method." Applied Sciences, 15(1), 2025, 123.

[Popoola et al., 21] Popoola, S. I., Adebisi, B., Ande, R., Hammoudeh, M., Anoh, K., Atayero, A. A.: "SMOTE-DRNN: A deep learning algorithm for botnet detection in the Internet-of-Things networks." Sensors, 21(9), 2021, 2985.

[Pradipta et al., 21] Pradipta, G. A., Wardoyo, R., Musdholifah, A., Sanjaya, I. N. H., Ismail, M.: "SMOTE for handling imbalanced data problem: A review." In Proc. 6th Int. Conf. Informatics and Computing (ICIC), 2021, 1–8. IEEE.

[Saleh et al., 24] Saleh, R. A., Al-Areqi, F., Konyar, M. Z., Kaplan, K., Öngir, S., and Ertunc, H. M. "Advancingtire safety: Explainable artificial intelligence-powered foreign object defect detection with xception networks and grad-cam interpretation." Applied Sciences, 14(10), 2024, 4267.

[Sayegh et al., 24] Sayegh, H. R., Dong, W., Al-Madani, A. M.: "Enhanced intrusion detection with LSTM-based model, feature selection, and SMOTE for imbalanced data." Appl. Sci., 14(2), 2024, 479.

[Soe et al., 19] Soe, Y. N., Santosa, P. I., Hartanto, R.: "DDoS attack detection based on simple ANN with SMOTE for IoT environment." In Proc. 4th Int. Conf. Informatics and Computing (ICIC), 2019, 1–5. IEEE.

[Tasnim et al., 22]Tasnim, A., Hossain, N., Parvin, N., Tabassum, S., Rahman, R., Hossain, M. I.: "Experimental analysis of classification for different Internet of Things (IoT) network attacks using machine learning and deep learning." In Proc. 2022 International Conference on Decision Aid Sciences and Applications (DASA), 2022, 406–410. IEEE.

[Taud and Mas, 17] Taud, H., Mas, J.-F.: "Multilayer perceptron (MLP)." In Geomatic Approaches for Modeling Land Change Scenarios, 2017, 451–455. Springer.

[Venkatasubramanian et al., 25] Venkatasubramanian, S., Pradeep Kumar Ch, S. K. B., Babu, B. P.: "Overcoming dataset imbalances and computational challenges in IoT intrusion detection: A SMOTE-enhanced transformer-based model." In Proc. 2025 IEEE 14th International Conference on Communication Systems and Network Technologies (CSNT), 2025, 1195–1204. IEEE.

[Verma and Ranga, 20] Verma, A., Ranga, V.: "Machine learning based intrusion detection systems for IoT applications." Wireless Pers. Commun., 111(4), 2020, 2287–2310.

[Zhang et al., 22] Zhang, P., Jia, Y., Shang, Y.: "Research and application of XGBoost in imbalanced data." Int. J. Distrib. Sens. Netw., 18(6), 2022, 15501329221106935.