Quantifying Feature Impact on IoT Intrusion Detection Performance in Imbalanced Botnet Datasets

Luis David Garcia
Computer Science
California Polytechnic State University
San Luis Obispo, USA
lgarc120@calpoly.edu

Nicholas Zarate
Computer Science
California Polytechnic State University
San Luis Obispo, USA
nezarate@calpoly.edu

***Abstract*—The widespread adoption of the Internet of Things (IoT) across sectors such as energy, healthcare, and automotive enhances productivity but introduces significant security vulnerabilities, notably to Denial of Service (DoS) and Distributed Denial of Service (DDoS) attacks. This study investigates the impact of dataset imbalance on the efficacy of machine learning models in detecting such security threats. By applying the Synthetic Minority Over-sampling Technique (SMOTE) to adjust the training data distribution from a 60/40 to both 50/50 and 80/20 benign-to-DDoS/DoS splits, we evaluate the performance improvements in detection models. Our results show that the XGBoost model, under these adjusted distributions, achieved an exceptional F1-score of 0.999983. Furthermore, random forest feature selection proved optimatl to find the top ten features. This research underscores the importance of dataset balance in enhancing the robustness of IoT security detection models, offering insights for future advancements in the field.**

***Index Terms*—Intrusion Detection Systems, IoT Security, Botnet Detection, DDoS Attacks, Feature Selection, Machine Learning, Deep Learning, Cybersecurity, Network Traffic Analysis**

I. Introduction

The digital landscape is increasingly besieged by sophisticated cyber threats, with botnet-driven Distributed Denial of Service (DDoS) attacks emerging as a formidable challenge to the stability of Internet services. The pervasive nature of these attacks underscores the critical need for effective Intrusion Detection Systems (IDS) capable of safeguarding IoT networks. Traditional defense mechanisms predominantly rely on static rule and signature-based IDS to counteract well-identified botnets, such as the Mirai botnet of 2016, by inspecting network flow data from devices like routers [1].

```
alert tcp any any -> any $HTTP_PORTS
(msg:"MALWARE-CNC User-Agent known malicious
user-agent string - Mirai"; flow:to_server,
established; content:"User-Agent|3a| Hello,
world"; nocase; http_header;
classtype:trojan-activity; sid:58992; rev:1;)
```

Fig. 1: Snort Rule for Detecting Mirar Botnet Based on SID.

An example of this approach is depicted in Figure 1, illustrating a Snort rule designed to detect the Mirai botnet based on a unique Signature ID (SID 1:58992). This rule targets a specific ”Hello, world” message characteristic of Mirai’s early infection stage. However, the reliance on fixed signatures renders these systems vulnerable to novel, or Zero-Day, attacks that deviate from recognized patterns [2]. The inadequacy of static rules in the face of evolving threats has propelled research towards the integration of machine learning (ML) and deep learning (DL) techniques, aiming to enhance predictive capabilities in IDS [3]. Yet, the adoption of such models faces skepticism due to their ”black box” nature, which obscures the decision-making process, undermining trust among security professionals [4].

This research endeavors to bridge the gap between the advanced detection capabilities of artificial intelligence and the need for transparency within intrusion detection with the following contributions:

- **Attack Analysis Exploration (Section 4):** An analysis of CIC-IDS-2018 dataset’s DoS and DDoS attack behaviors in IoT networks, aiming to refine IDS capabilities by closely examining attack signatures and patterns.
- **Model Performance Evaluation (Section 4):** An assessment of ML DoS detection across balanced and imbalanced datasets, using common metrics such as accuracy, precision, recall, and F1 score to gauge performance.
- **Balancing Strategy Analysis (Section 4):** The effectiveness of dataset balancing techniques, such as SMOTE and traditional undersampling, is evaluated to determine their impact on the predictive accuracy of IDS models.

We first delve into the existing studies [Section 2] then by clarifying the mechanisms underlying IDS predictions, we assist security engineers in crafting a more robust threat landscape for their organizational models [Section 3].

We acknowledge the constraints of time and resources, which have limited our ability to conduct an exhaustive comparative analysis of models and perform dataset balancing. The discussion of these limitations is critical for the scope and

implications for future research [Sections 5 and 6].

II. Background and Related Work

A. IoT Security and Botnet/DDoS Attacks

IoT networks are increasingly targeted by DDoS attacks, often initiated by botnets, which are networks of infected devices controlled by attackers [5]. These devices generate excessive traffic, leading to service disruption. A study by Trustwave in 2023 reported that botnets were behind over 95% of malicious web traffic, with around 19% of all web traffic being malicious [6].

B. Intrusion Detection Strategies and Challenges

To defend against these attacks, intrusion detection systems (IDS) employ methods like Random Forest for feature selection, improving the detection of potential threats [13]. However, the prevalence of benign activities within datasets can hide the presence of attacks, affecting the accuracy of IDS [14]. Effective feature selection is essential to counteract this, as traditional reliance on IP addresses is not sufficient due to their susceptibility to spoofing [15].

C. Research Gap

While there have been advancements in this field, understanding the impact of feature selection and dataset balance on IDS performance in IoT remains incomplete. This study seeks to bridge this gap, providing insights into the optimization of IDS for better security in the IoT domain.

D. Related Work

This section highlights key studies that influenced our choice of Random Forests for feature selection on the CIC-IDS-2018 dataset [16]. The focus was on packet-based metrics, flow statistics, and connection behaviors, reflecting network traffic's complexity. These studies underscore the necessity for a strategic approach to identify the most discriminative features for distinguishing between benign and malicious patterns. While Random Forests demonstrate a balance between efficiency and effectiveness, challenges such as feature importance variability remain, guiding our research towards improving this methodology [17]

Genetic Algorithms: Utilize evolutionary processes to pinpoint optimal features for intrusion detection, though their inherent randomness may affect consistency in accuracy [18].

Random Forests: Known for pinpointing crucial feature importance, they play a significant role in the domain. However, their performance may vary due to the stochastic nature of their underlying decision trees [19].

XGBoost: As a leading gradient boosting method, XGBoost offers enhanced model accuracy and efficiency. Yet, the complexity involved in hyperparameter tuning necessitates a comprehensive grasp of its mechanics, presenting a challenge [19].

Decision Trees: Fundamental to more sophisticated models like Random Forests and XGBoost, they offer a clear, interpretable decision-making process. Despite their advantages, there's a risk of overfitting with complex data [19].

Logistic Regression: This statistical method shines in binary classification with linearly related variables but faces limitations with non-linear relationships [19].

Spearman's Rank Correlation Coefficient: Effective in identifying linearly related features but limited in addressing complex, non-linear relationships [20].

Sequential Backward Selection (SBS): Focuses on eliminating less impactful features methodically, yet its computational intensity may not suit large datasets [21].

Pearson Correlation Filter and WCC Optimization Algorithm: Aims to refine the feature selection process, though it struggles with highly correlated features, resulting in redundancy [22].

This comparative analysis, while recognizing the constraints of each method, guided our choice of Random Forest for feature selection within the CIC-IDS-2018 dataset [23]. This decision reflects a strategic balance between methodological thoroughness and applicability, particularly for complex and imbalanced intrusion detection data. Our future research will focus on harnessing Random Forest's advantages while mitigating its limitations, aiming to boost the efficacy of feature selection for botnet/DDoS attack detection in IoT networks.

III. Methodology

When working with classification datasets, the issue of an imbalanced dataset is when there is a disproportionate number of instances across classes, with one class notability being more prevalent than the others. This problem is not only a concern to binary classification but also extends to multi-class data, the latter being what we are interested in our study. Issues arise when training models on imbalanced datasets due to the bias towards the over-represented class. This becomes problematic in cases in which we are interested in accurate classification of the minority classes. Lack of sufficient data for the minority classes can lead to poor learning of their characteristics which affects the predictive accuracy of the model in multi-class applications. To explore if varying levels of class imbalance within a dataset affect the performance of machine learning models, we developed a framework for comparison to facilitate this. The flowchart outline of the framework's procedure is visualized in Figure 3. Furthermore, considering that there is a substantial amount of features per entry, we have also included a step to perform feature selection to investigate if a reduced amount of features has a noticeable impact on model performance. The specifics of the dataset used for training, data preprocessing, employed models, and performance metrics will be elaborated upon in the subsequent sections.

A. Dataset Used

In this research study, the CSE-CIC-IDS2018 dataset was used and applied to classify network traffic. The dataset was collaboratively created by the Communications Security Establishment (CSE) and The Canadian Institute for Cybersecurity (CIC). This dataset was selected due to it being publically available, its substantial and recent status of intrusion detection

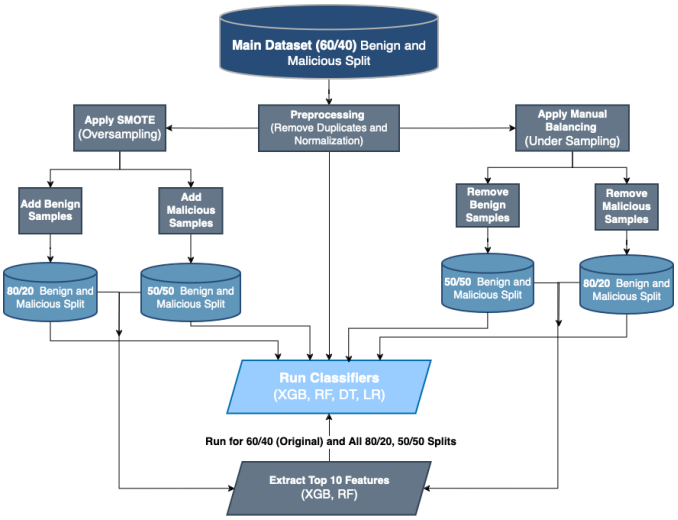

Fig. 2: System Architecture Framework for Classifying and Comparison.

data, and its covering of multiple malicious attack types, including those of particular interest in our study, such as DoS (Denial of Service) and DDoS (Distributed Denial of Service). The dataset in total has 16,233,002 instances that span over ten days. The distribution of the dataset is as follows: Benign (83.07%), DDoS (7.786%), DoS (4.031%), Brute-force (2.347%), Botnet (1.763%), Infiltration (0.997%) and Web attack (0.006%) [25].

B. Dataset pre-processing

Due to limited computational power to handle the complete dataset, we decided to explore day 15 which was comprised of 95% benign data and 5% DoS attacks-GoldenEye and DoS attacks-Slowloris as the attack types. This subsection of the dataset initially contained 1,048,575 rows with 80 features. To mitigate the overfitting of the models from the start, we added more DoS and DDoS data from other days to the day 15 dataset. DoS attacks-Hulk and DoS attacks-SlowHTTPTest was added from day 16 and DDOS attack-HOIC was added from day 21. This curated dataset is now comprised of 2,968,430 samples, 1,801,894 being benign and 1,174,536 being the attack types. Preprocessing the dataset before building the machine learning model is crucial to ensure that the data is represented in a format that the algorithm can understand. This helps prevent errors in the workflow that can negatively affect the performance of the model. Features in the dataset that do not contain unique values are removed as they do not provide useful information for the model to learn from. This corresponded with the following features being dropped: Bwd PSH Flags, Fwd URG Flags, Bwd URG Flags, CWE Flag Count, Fwd Byts/b Avg, Fwd Pkts/b Avg, Fwd Blk Rate Avg, Bwd Byts/b Avg, Bwd Pkts/b Avg, Bwd Blk Rate Avg. Subsequently, data that contains duplicates and excessively large or undefined values were deleted from the dataset. Furthermore, the timestamp feature was also removed as we were concerned with the models overfitting to specific times which makes them less generalizable. Data normalization was performed using Z-score standardization. Non-numeric values in the dataset like the target label were encoded before training the model. The final preprocessed dataset had 69 features and was comprised of 2,968,408 instances, 1,793,872 being benign and 1,174,536 being the attack types. The dataset was randomly split into a 70% training and a 30% testing set scheme for the machine learning algorithms.

X_train: Contains the features used for training, excluding the target label.

Y_train: Contains the target labels corresponding to X_train.

X_test: Contains the features for testing, also excluding the target label.

Y_test: Contains the target labels for evaluation, corresponding to X_test.

C. Machine learning classifiers

We employed four machine learning models in our study: Random Forest, Decision Tree, Logistic Regression, and XGBoost. These models were chosen due to being widely used and for their approaches to classification problems, however, each of them can be impacted by an imbalanced dataset. This is an important area to explore especially in the field of intrusion detection.

- **Random Forest and Decision Trees:** Both of these models can be affected by imbalanced datasets because they rely on criteria like information gain or Gini impurity for splitting at a node. These can become biased towards the majority class because they use class distribution information to help determine splits. This can influence the effectiveness of classifying the minority class.
- **Logistic Regression:** This model can be impacted by class imbalances because it tries to maximize the log-likelihood of the observed labels which can lead to it biasing the majority class in its calculation.
- **XGBoost:** Extreme Gradient Boosting is generally more robust to class imbalance but it can still be affected. The objective function of the model is to minimize prediction error, however, the error can be skewed by the majority class. This can lead to the model's predictions underrepresenting the minority class.

D. Feature Selection

For feature selection, we used the random forest classifier to identify and rank features with the most importance in the dataset. This enabled us to identify the top 10 influential features on the model's decision-making process. Feature selection was only applied to the original post-processed dataset to allow comparison of performance metrics across the experiments and to not bring in other sources of error.

E. Data Balancing

One of the primary aspects of our framework was assessing the effects of varying class balances on model performance versus the original processed dataset. Two strategies were employed:

1) Manual Balancing:

- **50/50 Balanced Dataset:** A subset of the benign data was randomly sampled to match the number of attack data records, ensuring an equal representation of both classes. This helps in mitigating the model's bias towards the more frequently occurring class.
- **80/20 Balanced Dataset:** This approach aimed to maintain a ratio where 80% of the data represents the benign class and 20% the malicious types. To achieve this desired balance, attack types were randomly sampled and dropped.

2) SMOTE Balancing: Manual balancing of a dataset usually involves undersampling the minority class by random selection which leads to loss of information and possibilities of overfitting. Therefore we address this issue by using Synthetic Minority Over-sampling Technique (SMOTE). SMOTE generates synthetic data samples for the minority class based on the existing samples within the dataset.

- **50/50 Balanced Dataset:** This approach balances the dataset to an equal distribution of benign to attack types. Synthetic data was generated and over-sampled from the minority attack class to match the total number of samples in the benign class.
- **80/20 Balanced Dataset:** In this case, the dataset is balanced to have 80% benign data and 20% attack data. Synthetic data was generated and over-sampled from the majority benign class to match to achieve this ratio.

F. Performance Metrics

Selecting the correct performance metrics is essential to effectively evaluating machine learning algorithms. This study focuses on four primary metrics: accuracy (A), precision (P), recall (R), and F1-score (F1). These metrics will allow us to comprehensively compare the algorithms when applied to the intrusion detection datasets.

$$\text{Accuracy} = \frac{\text{TP + TN}}{\text{TP + TN + FP + FN}} \tag{1}$$

$$\text{Precision} = \frac{\text{True Positive (TP)}}{\text{True Positive (TP) + False Positive (FP)}} \tag{2}$$

$$\text{Recall} = \frac{\text{True Positive (TP)}}{\text{True Positive (TP) + False Negative (FN)}} \tag{3}$$

$$\text{F1 Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision + Recall}} \tag{4}$$

Accuracy measures the overall correctness of the predictions made by the model. Precision reflects the proportion of correct positive predictions among all positive predictions. On the other hand, recall is the proportion of actual positives that are correctly identified. Lastly, the F1 score is derived as the harmonic mean of precision and recall, which is useful in evaluating incorrect classifications to give a more balanced view of the model's performance. Additionally, we report the training times in seconds of the algorithms to provide a well-rounded analysis. Within each type of balancing scheme, manual or SMOTE, and within each of the dataset ratios, the best-performing model based on the largest value will be boldfaced for each metric. The fastest training time will also be boldfaced.

IV. Results and Discussion

After performing feature selection to identify the top 10 features on the original preprocessed dataset, these key features emerged as contributing to detecting DoS and DDoS attacks. In order of importance, the rankings are as follows: Init Fwd Win Byts, Fwd Header Len, Fwd Seg Size Min, Dst Port, Fwd IAT Min, Subflow Bwd Pkts, Bwd Pkts/s, Pkt Size Avg, Bwd Header Len, Fwd IAT Tot.

A. Threat Analysis of Attacks in Dataset

This discussion delves further into the operational mechanisms and detection strategies of several notable DDoS tools. The CIC-IDS-2018 dataset's inclusion of DoS Goldeneye, Slowloris, SlowHTTPTest, Hulk, and DDoS HOIC attacks underlines the significance of these methods in the evolving narrative of cyber threats [7] [8].

The following sections provide an in-depth look at each of these attack tools, emphasizing the critical need for robust detection and mitigation techniques to safeguard against such pervasive threats.

- **DoS attacks - GoldenEye:** The GoldenEye tool orchestrates an application layer HTTP attack designed to monopolize all available sockets on an HTTP/S server. It leverages KeepAlive and Cache-Control options to maintain socket connections indefinitely, functioning independently of the operating system [9]. Detection strategies revolve around identifying unusual increases in HTTP connections and a surge in packet transmission rates.
- **DoS attacks - Slowloris:** Slowloris, identifiable as both a DoS and DDoS tool, specifically targets web servers using a single computer by focusing on port 80. This method gradually depletes server resources by initiating and keeping open partial HTTP connections [9]. Detection involves monitoring several parameters, including the number of connections from single IP addresses, the server's user load, HTTP connection counts, durations of user connections, and data transfer velocities.
- **DoS attacks - SlowHTTPTest:** This tool performs a slow HTTP DoS attack by consuming web server resources through the transmission of numerous incomplete HTTP requests. After establishing a TCP connection, it delays the finalization of HTTP requests, effectively clogging the server's connection queue as it awaits completion of the requests [10]. Detection primarily depends on spotting a rise in incomplete HTTP request numbers, especially by examining their frequency and, where possible, the structure of the request, typically ending with a "\r\n\r\n" sequence in GET requests.

- **DoS attacks - Hulk:** Dubbed the "HTTP Unbearable Load King," the Hulk tool executes a DDoS attack by inundating a web server with requests for multiple URLs from several sources, thereby masking its attack pattern [11]. Observing a sudden increase in network traffic, including a rapid escalation in the flow of bytes and packets per second, aids in its detection.
- **DDoS attack - HOIC:** The High Orbit Ion Cannon (HOIC) is developed to launch HTTP flood attacks by generating concurrent requests, incorporating TLS and SOCKS proxy support to disguise the attack traffic [12]. Detecting such attacks involves noting alterations in HTTP request patterns for GET and POST methods and the acceleration of packet rates.

Through an exploration of these sophisticated DDoS tools and their detection mechanisms, the necessity for advanced defensive strategies against the backdrop of an increasingly perilous cyber threat environment is further accentuated.

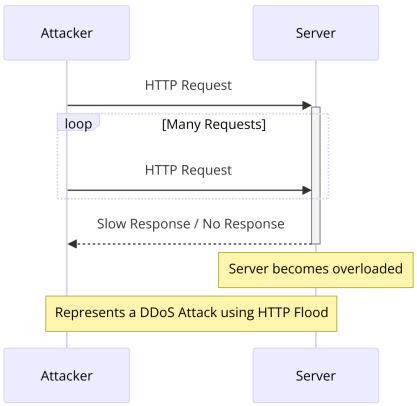

Fig. 3: HTTP Attack Flow Diagram.

Figure 3 depicts an HTTP flood attack, where an attacker sends numerous HTTP requests to a server in a short span of time. This overwhelming flood of requests leads to the server becoming overloaded, resulting in slow responses or no response at all, effectively rendering the service unavailable to legitimate users.

B. Data Distribution Analysis

The data distrubtion analysis here considers three distinct datasets. The 60/40 Original Dataset serves as the foundational benchmark with its intrinsic data distributions. Subsequently, we evaluate the 80/20 and 50/50 Balanced Datasets created via SMOTE and limited to the top 10 features. This focus is a result of these tailored datasets having superior machine learning model performance when contrasted with those derived from manual balancing techniques.

1) Analysis of the 60/40 Original Dataset: The 60/40 Original Dataset Table 1 displays a disproportionate representation of benign samples, comprising 60.5% of the data. This imbalance reflects a common challenge in cybersecurity datasets, where normal traffic significantly outnumbers attack instances.

Dataset Condition	60/40 Original Dataset
Benign Samples	1,801,894
DDOS-HOIC	668,461
GoldenEye	41,455
Hulk	434,873
SlowHTTPTest	19,462
Slowloris	10,285
Total Samples	2,968,408
Features	69

TABLE I: Distribution of the 60/40 Original Dataset

However, the data does include a considerable variety of attack types, with DDOS-HOIC being the most prevalent attack category at approximately 22.5%. The variability of attack types, alongside a high feature count (69), suggests a complex dataset potentially beneficial for training models to distinguish subtle patterns indicative of different attacks. However, the imbalance could bias the model towards predicting benign samples more frequently, possibly at the expense of detection sensitivity.

Dataset Condition	80/20 Balanced w/ SMOTE
Benign Samples	1,137,137
DDOS-HOIC	142,938
GoldenEye	30,043
Hulk	104,208
SlowHTTPTest	7,056
Slowloris	37
Total Samples	1,421,419
Features	11

TABLE II: Distribution of the 80/20 Balanced Dataset with SMOTE

2) Analysis of the 80/20 Balanced Dataset with SMOTE: The 80/20 Balanced Dataset with SMOTE (Table 2) shows a significant reduction in benign samples compared to the original dataset, which is a result of the resampling technique applied to balance the classes. The SMOTE algorithm has augmented the minority classes to reach an 80/20 distribution. The balancing process, while reducing the number of features to 11, may have mitigated the risk of model bias towards benign samples. However, the reduced feature set could also limit the model's ability to capture the complexity of attack patterns, which is critical for effective anomaly detection. The lower variability in attack representation compared to the original dataset might streamline the model training process but could also risk overfitting to the resampled data.

Dataset Condition	50/50 Balanced w/ SMOTE
Benign Samples	1,137,137
DDOS-HOIC	571,752
GoldenEye	120,173
Hulk	416,832
SlowHTTPTest	28,225
Slowloris	151
Total Samples	2,274,270
Features	11

TABLE III: Distribution of the 50/50 Balanced Dataset with SMOTE

3) Analysis of the 50/50 Balanced Dataset with SMOTE: The 50/50 Balanced Dataset with SMOTE (Table 3) demonstrates an even distribution between benign and attack samples, which could potentially eliminate the class imbalance problem present in the original dataset. The attack categories are now represented more equally, with DDOS-HOIC still being the predominant attack but less so than in the original dataset. This balance allows for a potentially more robust model training, where the model can learn to identify attacks without an inherent bias towards benign classifications. However, the balancing act achieved through SMOTE might introduce synthetic artifacts that do not represent true attack behaviors, possibly leading to a model that generalizes less effectively to real-world data. Additionally, similar to the 80/20 dataset, the reduction to 11 features might not capture the full complexity of attack signatures.

In conclusion, the SMOTE resampling technique has adjusted the class distribution significantly, which can aid in model development by addressing class imbalance. Nonetheless, the reduction in features in the balanced datasets calls for a cautious interpretation of model performance, especially in regards to the generalization capabilities beyond the resampled dataset. Furthermore, it is crucial to consider the implications of synthetic sample generation and feature selection when applying the trained models to real-world scenarios, where diverse and complex attack patterns are expected.

C. Feature Analysis

This study closely examines several key features instrumental in detecting DoS and DDoS attacks within network traffic, revealing insights into their operational significance and implications for network security management.

Initial Forward Window Bytes suggests tactics like window size manipulation, a common approach in these cyber attacks aimed at disrupting network flow management.

Forward Header Length and **Backward Header Length** provide insights into packet tampering and unusual packet sizes, often markers of fraudulent activity and packet manipulation.

Minimum Size of Forward Segment and **Average Packet Size** underscore the strategic alterations in packet sizes, potentially signaling attempts to disturb the established traffic patterns.

Destination Port is pivotal for identifying ports frequently targeted by attackers, serving as a gateway for intrusion activities.

Temporal attributes, notably **Minimum Inter-Arrival Time** and **Total Inter-Arrival Time of Forward Packets**, highlight the irregular timing patterns inherent in DoS/DDoS traffic, essential for temporal analysis in intrusion detection.

Subflow Backward Packets and **Backward Packets per Second** illustrate abnormal response patterns and elevated traffic rates in response to malicious queries, indicative of a network under siege.

Collectively, these features furnish a comprehensive picture of network behavior amidst DoS/DDoS attacks, underscoring their significance in bolstering intrusion detection systems. Through meticulous analysis, this study elucidates the critical role these features play in identifying and mitigating cyber threats, thereby enhancing network security posture.

D. Model Performance Analysis

This analysis delves into the impact of data balancing techniques on the performance of various machine learning models, specifically focusing on their ability to detect DoS and DDoS attacks. Through a series of experiments, encapsulated in multiple tables, the study meticulously evaluates how manual balancing, SMOTE, and feature reduction influence model efficacy.

XGBoost's Superiority Across Conditions: Across diverse dataset conditions, XGBoost consistently emerges as the most effective model, particularly excelling in balanced scenarios. Table 4 highlights XGBoost's robust performance, with notable F1-Scores of 0.99995 and 0.99994 in 50/50 and 80/20 balanced datasets, respectively, surpassing Random Forest's performance in the original dataset configuration. This demonstrates XGBoost's enhanced sensitivity to balanced data distributions, crucial for the nuanced detection of intrusion attempts.

Feature Reduction's Impact on Efficiency: The decision to limit the feature set to the top 10 significantly impacts model training efficiency, as seen in Table 5. This reduction not only shortens training times dramatically, with Decision Trees clocking in at a mere 1.59 seconds but also maintains high F1-Scores for XGBoost, underscoring the effectiveness of targeted feature selection in optimizing model performance without sacrificing detection capabilities.

SMOTE Oversampling's Role in Performance Enhancement: Table 6 illustrates the efficacy of SMOTE oversampling in enhancing model performance, especially for XGBoost in the 50/50 and 80/20 balanced scenarios. This technique's ability to generate synthetic samples results in a more nuanced model training process, allowing for a more accurate representation of minority classes and thereby boosting model performance in detecting sophisticated attack patterns.

Advantages of a Reduced Feature Set: Table 7 further reinforces the benefits of a streamlined feature set. With only the top 10 features, XGBoost continues to outperform across all metrics, achieving high F1-Scores while significantly reducing training times. This approach not only makes model training more efficient but also reduces the computational resources required, making it a practical solution for real-world applications.

Highlight on Real-World Application Potential: A particularly promising finding is XGBoost's performance with the manually balanced 50/50 dataset using only the top 10 features. With an impressive F1-Score of 0.99988 and a training time of just 44.65 seconds, this scenario presents a viable, efficient approach for intrusion detection systems. It showcases the potential for deploying high-accuracy models that are both time and resource-efficient, a critical consideration in the operational environments of IoT networks.

E. Summary of Key Results

The comparative evaluation of four machine learning models across datasets of varied class distributions and feature counts reveals a distinct performance hierarchy. XGBoost emerges as the superior model, demonstrating exemplary accuracy and F1 scores across all conditions, including original, 50/50 balanced, and 80/20 balanced datasets. This consistent performance holds true regardless of the data balancing technique employed, as evidenced in Tables 4, 5, 6, and 7. Notably, XGBoost also ranks as the second-fastest model, trailing only behind Decision Trees in terms of computational speed. Its efficacy in navigating both imbalanced and balanced datasets underscores its robustness and adaptability for a range of intrusion detection scenarios.

Although Random Forest exhibits commendable performances, it falls short of XGBoost's high benchmark. Logistic Regression and Decision Trees display competence in specific contexts but tend to lag in more complex or nuanced scenarios. This discrepancy highlights potential limitations in their ability to process intricate data patterns, especially within balanced datasets or when analyzing a reduced set of features.

This study conclusively demonstrates the critical role of data balancing and feature reduction in optimizing machine learning model outputs for intrusion detection tasks. The findings strongly endorse the implementation of XGBoost, paired with meticulous data preprocessing methods, to enhance detection precision and maximize operational efficacy. Through this analytical journey, the study contributes valuable insights into advancing machine learning applications within the domain of cybersecurity.

TABLE IV: Performance by Dataset Ratio: Manual Balancing

Model	Accuracy	Precision	Recall	F1-Score	Time (s)
Original Dataset					
RF	0.99998	**0.99993**	0.99988	**0.99991**	575.57
DT	0.99998	0.99991	0.99973	0.99982	**80.25**
LR	0.99867	0.99405	**0.99891**	0.99646	551.40
XGB	**0.99999**	0.99986	0.99986	0.99986	197.70
50/50 Balanced					
RF	0.99999	**0.99997**	0.99990	0.99993	495.90
DT	0.99997	0.99977	0.99971	0.99974	**53.16**
LR	0.99882	0.99527	0.99885	0.99705	428.31
XGB	**1.00000**	0.99996	**0.99995**	**0.99995**	123.45
80/20 Balanced					
RF	0.99992	0.99960	0.99956	0.99958	479.30
DT	0.99998	0.99973	0.99953	0.99963	**51.28**
LR	0.99851	0.98762	0.99743	0.99242	418.53
XGB	**1.00000**	**0.99999**	**0.99990**	**0.99994**	93.91

TABLE V: Top 10 Features: Manual Balancing Performance

Model	Accuracy	Precision	Recall	F1-Score	Time (s)
Original Dataset					
RF	0.99997	0.99977	0.99985	0.99981	128.45
DT	0.99993	0.99947	0.99936	0.99942	**4.52**
LR	0.99550	0.86078	0.84069	0.84486	286.19
XGB	**0.99998**	**0.99982**	**0.99991**	**0.99987**	70.36
50/50 Balanced					
RF	**0.99992**	0.99957	0.99989	0.99973	41.46
DT	0.99984	0.99942	0.99970	0.99956	**1.59**
LR	0.99462	0.90710	0.83159	0.83968	109.16
XGB	**0.99992**	**0.99985**	**0.99991**	**0.99988**	44.65
80/20 Balanced					
RF	**0.99997**	**0.99978**	**0.99985**	**0.99982**	114.52
DT	0.99993	0.99952	0.99952	0.99952	**3.35**
LR	0.99558	0.81424	0.82093	0.81754	250.15
XGB	**0.99997**	**0.99978**	0.99981	0.99979	226.53

TABLE VI: Performance by Dataset Ratio with SMOTE

Model	Accuracy	Precision	Recall	F1-Score	Time (s)
Original Dataset					
RF	0.99998	**0.99993**	0.99988	**0.99991**	529.39
DT	0.99998	0.99991	0.99973	0.99982	**70.73**
LR	0.99867	0.99405	**0.99891**	0.99646	541.74
XGB	**0.99999**	0.99986	0.99986	0.99986	123.08
50/50 Balanced with SMOTE					
RF	0.99998	0.99984	0.99985	0.99985	622.14
DT	0.99999	0.99990	0.99994	0.99992	**40.74**
LR	0.99868	0.99385	0.99934	0.99656	637.93
XGB	**1.00000**	**1.00000**	**0.99999**	**0.99999**	138.92
80/20 Balanced with SMOTE					
RF	0.99997	**0.99995**	0.99985	0.99990	714.00
DT	**0.99999**	0.99986	0.99990	0.99988	**91.58**
LR	0.99869	0.99375	0.99890	0.99629	649.32
XGB	**0.99999**	**0.99995**	**0.99993**	**0.99994**	151.21

TABLE VII: Performance with SMOTE and Top 10 Features

Model	Accuracy	Precision	Recall	F1-Score	Time (s)
Original Dataset					
RF	**0.99997**	0.99977	0.99985	0.99981	119.30
DT	0.99993	0.99947	0.99936	0.99942	**3.67**
LR	0.99550	0.86078	0.84069	0.84486	260.73
XGB	0.99998	**0.99982**	**0.99991**	**0.99987**	187.30
50/50 Balanced with SMOTE					
RF	0.99998	0.99986	0.99993	0.99989	213.80
DT	0.99997	0.99984	0.99978	0.99981	**5.96**
LR	0.99519	0.93691	0.98982	0.96031	430.52
XGB	**0.99998**	**0.99986**	**0.99987**	**0.99987**	143.18
80/20 Balanced with SMOTE					
RF	0.99997	0.99975	**0.99991**	0.99983	121.03
DT	0.99994	0.99966	0.99946	0.99956	**3.61**
LR	0.99568	0.87029	0.84082	0.84625	242.61
XGB	**0.99998**	**0.99982**	0.99986	**0.99984**	122.87

V. Conclusion and Future Work

In evaluating the impact of class distribution and balancing techniques on model efficacy, our analysis establishes XGBoost as the top performer across various datasets, with Random Forest also showing robust results albeit with longer training times. The limited scope of our test dataset, however, suggests that further studies should utilize a broader range of data to confirm these outcomes. Future research should aim to expand the dataset diversity for a more robust evaluation and consider the increased computational requirements this entails. The investigation into the top 10 features has provided key insights, paving the way for tailored intrusion detection systems optimized for DoS/DDoS scenarios.

Looking forward, we see critical pathways for advancing IoT security against botnet and DDoS attacks: comparing machine learning methodologies, exploring IoT-specific features with Explainable AI, developing resource-efficient models, and employing advanced statistical validations. Progressing these fronts will contribute substantially to the resilience of digital ecosystems against cyber threats.

VI. Ethic Statement

It is imperative to acknowledge the ethical implications of developing and deploying machine learning models for DDoS and DoS detection. Our research does not explore the potential weaknesses of these models, and we discourage the use of open-source tools for malicious activities. Adequate safeguards should be in place to protect these models from adversarial attacks. Furthermore, the environmental cost of training these models must be considered, with strategies to reduce their ecological footprint.

Acknowledgment

Our heartfelt thanks go to Dr. Dongfeng Fang for her unwavering support and assistance throughout our project. We are also grateful to California Polytechnic State University for providing us with the resources necessary to conduct our research.

References

[1] P. Kumari and A. K. Jain, “A Comprehensive Study of DDoS Attacks over IoT Network and Their Countermeasures,” Computers & Security, p. 103096, Jan. 2023, doi: https://doi.org/10.1016/j.cose.2023.103096.

[2] “Snort - Rule Docs,” snort.org. https://snort.org/rule_docs/1-58992 (accessed Feb. 25, 2024).

[3] R. Ahmad, I. Alsmadi, W. Alhamdani, and L. Tawalbeh, “Zero-day attack detection: a systematic literature review,” Artificial Intelligence Review, vol. 56, Feb. 2023, doi: https://doi.org/10.1007/s10462-023-10437-z.

[4] Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, “Electron spectroscopy studies on magneto-optical media and plastic substrate interface,” IEEE Transl. J. Magn. Japan, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].

[5] C. C. Editor, “Botnet - Glossary — CSRC,” csrc.nist.gov. Botnet Glossary Term

[6] P. Knapczyk and W. Cieslak, “SpiderLabs Blog — Trustwave,” www.trustwave.com, Jul. 05, 2023. Trustwave Article Research.

[7] Thakkar, Ankit, and Ritika Lohiya. “A Review of the Advancement in Intrusion Detection Datasets.” Procedia Computer Science, vol. 167, 1 Jan. 2020, pp. 636–645, www.sciencedirect.com/science/article/pii/S1877050920307961, https://doi.org/10.1016/j.procs.2020.03.330.

[8] “IDS 2018 — Datasets — Research — Canadian Institute for Cybersecurity — UNB,” www.unb.ca. https://www.unb.ca/cic/datasets/ids-2018.html

[9] T. Shorey, D. Subbaiah, A. Goyal, A. Sakxena and A. K. Mishra, ”Performance Comparison and Analysis of Slowloris, GoldenEye and Xerxes DDoS Attack Tools,” 2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI), Bangalore, India, 2018, pp. 318-322, doi: 10.1109/ICACCI.2018.8554590.

[10] N. Tripathi, N. Hubballi and Y. Singh, ”How Secure are Web Servers? An Empirical Study of Slow HTTP DoS Attacks and Detection,” 2016 11th International Conference on Availability, Reliability and Security (ARES), Salzburg, Austria, 2016, pp. 454-463, doi: 10.1109/ARES.2016.20.

[11] Hassen, Oday A., and H. Ibrahim. ”Preventive Approach against HULK Attacks in Network Environment.” 2017, International Journal of Computing and Business Research 7.3.

[12] S. Black and Y. Kim, ”An Overview on Detection and Prevention of Application Layer DDoS Attacks,” 2022 IEEE 12th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, USA, 2022, pp. 0791-0800, doi: 10.1109/CCWC54503.2022.9720741.

[13] B. Kaur et al., “Internet of Things (IoT) security dataset evolution: Challenges and future directions,” Internet of Things, p. 100780, Apr. 2023, doi: https://doi.org/10.1016/j.iot.2023.100780.

[14] M. Mittal, K. Kumar, and S. Behal, “Deep learning approaches for detecting DDoS attacks: a systematic review,” Soft Computing, Jan. 2022, doi: https://doi.org/10.1007/s00500-021-06608-1.

[15] J. Liu, M. Simsek, B. Kantarci, M. Bagheri and P. Djukic, ”Collaborative Feature Maps of Networks and Hosts for AI-driven Intrusion Detection,” GLOBECOM 2022 - 2022 IEEE Global Communications Conference, Rio de Janeiro, Brazil, 2022, pp. 2662-2667, doi: 10.1109/GLOBECOM48099.2022.10000985.

[16] “A Realistic Cyber Defense Dataset (CSE-CIC-IDS2018) - Registry of Open Data on AWS,” registry.opendata.aws. https://registry.opendata.aws/cse-cic-ids2018/

[17] Nadim Elsakaan, and Kamal Amroun. “A Comparative Study of Machine Learning Binary Classification Methods for Botnet Detection.” Lecture Notes in Networks and Systems, 1 Jan. 2022, pp. 20–34, https://doi.org/10.1007/978-3-030-95918-0_3. Accessed 21 Feb. 2024.

[18] X. Liu and Y. Du, “Towards Effective Feature Selection for IoT Botnet Attack Detection Using a Genetic Algorithm,” Electronics, vol. 12, no. 5, pp. 1260–1260, Mar. 2023, doi: https://doi.org/10.3390/electronics12051260.

[19] L. D’hooge, M. Verkerken, T. Wauters, F. De Turck, and B. Volckaert, “Investigating Generalized Performance of Data-Constrained Supervised Machine Learning Models on Novel, Related Samples in Intrusion Detection,” Sensors, vol. 23, no. 4, p. 1846, Feb. 2023, doi: https://doi.org/10.3390/s23041846.

[20] “Implementation of Ensemble Learning and Feature Selection for Performance Improvements in Anomaly-Based Intrusion Detection Systems — IEEE Conference Publication — IEEE Xplore,” ieeexplore.ieee.org. https://ieeexplore.ieee.org/abstract/document/9172014 (accessed Feb. 21, 2024).

[21] “In-Depth Feature Selection for the Statistical Machine Learning-Based Botnet Detection in IoT Networks — IEEE Journals & Magazine — IEEE Xplore,” ieeexplore.ieee.org. https://ieeexplore.ieee.org/abstract/document/9875261.

[22] Y. Masoudi-Sobhanzadeh and S. Emami-Moghaddam, “A real-time IoT-based botnet detection method using a novel two-step feature selection technique and the support vector machine classifier,” Computer Networks, vol. 217, p. 109365, Nov. 2022, doi: https://doi.org/10.1016/j.comnet.2022.109365.

[23] Ismail et al., ”A Machine Learning-Based Classification and Prediction Technique for DDoS Attacks,” in IEEE Access, vol. 10, pp. 21443-21454, 2022, doi: 10.1109/ACCESS.2022.3152577.

[24] Alatram, Alaa, et al. “DoS/DDoS-MQTT-IoT: A Dataset for Evaluating Intrusions in IoT Networks Using the MQTT Protocol.” Computer Networks, vol. 231, 1 July 2023, p. 109809, www.sciencedirect.com/science/article/pii/S1389128623002542, https://doi.org/10.1016/j.comnet.2023.109809 .

[25] J. L. Leevy and T. M. Khoshgoftaar, “A survey and analysis of intrusion detection models based on CSE-CIC-IDS2018 Big Data,” Journal of Big Data, vol. 7, p. 104, 2020, doi: https://doi.org/10.1186/s40537-020-00382-x.