Published by :
http://www.ijert.org

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14 Issue 06, June - 2025

Automated Identification and Forensic Analysis of Network Traffic Anomalies Through Ensemble Learning Techniques: An Advanced Machine Learning Frame Work for Cybersecurity Threat Intelligence

Emmanuel B Usoro,
Department of Cyber Security, Faculty of Computing, University of Uyo, Nigeria.

Edidiong M Etuk
Department of Computer Engineering, Faculty of Engineering, University of Uyo, Nigeria.

Abstract: **- In the era of increasing cyber threats, automatically identifying and analyzing anomalies within network traffic is essential for robust cybersecurity intelligence. This study explores the application of ensemble learning methods to enhance anomaly detection in network traffic. The dataset underwent thorough preprocessing and descriptive statistical analysis, confirming the proper normalization of key features. Among the ensemble models tested, AdaBoost achieved a strong overall accuracy of 0.89, with high precision (0.90) and recall (0.99) for normal traffic classification. XGBoost also performed effectively, with an accuracy of 0.88, showcasing its capability to analyze complex network behaviors. The proposed framework establishes a solid foundation for integrating intelligent systems into cybersecurity infrastructures, supporting proactive anomaly detection and in-depth forensic analysis.**

Keywords: **Machine Learning, Network, Threat, Anomaly detection, Forensic Analysis**

1.0 INTRODUCTION

In today's hyperconnected digital age, cybersecurity has become more than a technical necessity. It is a fundamental pillar of protecting personal data, corporate assets, and national infrastructure. As the world increasingly depends on the seamless flow of information, the security of networks that carry this information is under constant threat. With the growing sophistication of cyberattacks, especially those that are difficult to predict or identify early, many traditional security systems are struggling to remain effective. Systems that rely solely on predefined rules or known attack patterns, such as signature-based intrusion detection systems, often fail to detect newer and more complex threats that evolve rapidly (Firmansyah, 2024; Wang, 2024).

The complexity and volume of modern network traffic further compound this growing gap in effective threat detection. The huge amount of data flowing across networks every second makes it harder for conventional tools to detect abnormal behaviors in real-time. As attackers develop new ways to disguise their malicious activities within normal-looking traffic, cybersecurity professionals need smarter and more dynamic tools to keep up. Network forensics plays an essential role in this effort. It allows analysts to investigate and trace back unauthorized or suspicious activities by closely examining the flow and characteristics of traffic on a network (Paracha et al., 2024). Network traffic anomalies such as Distributed Denial of Service (DDoS) attacks, Man-in-the-Middle (MitM) intrusions, and Advanced Persistent Threats (APTs) pose serious challenges. These attacks often hide in plain sight and can cause severe disruption or data loss before being discovered. For example, DDoS attacks flood systems with traffic to bring them down, while MitM attacks secretly intercept data between users. APTs, on the other hand, involve long-term infiltration to quietly harvest sensitive information. Detecting such attacks demands techniques that go beyond traditional filters and firewalls and that can understand the deeper behavior of data patterns (Agarwal et al., 2023).

To counter these emerging threats, the cybersecurity field is increasingly embracing machine learning (ML). ML models are capable of learning from past behaviors to recognize suspicious activities that deviate from the norm. In particular, ensemble learning has gained popularity due to its ability to combine multiple models to deliver better results than a single algorithm. Unlike standalone models, ensemble techniques can integrate different perspectives, improving both accuracy and reliability in identifying subtle or complex anomalies hidden in large-scale network data. Methods like Random Forests and Gradient Boosting are especially effective in handling the high-dimensional, often unbalanced nature of network traffic data. What makes ensemble learning particularly powerful in cybersecurity is its ability to support forensic investigation, not just real-time threat detection. These models can offer detailed insight into how and where a breach occurred, helping analysts' piece together the timeline and origin of an attack. This deeper understanding is essential for developing stronger defense mechanisms and for reducing future risk. In addition, ensemble models are more resilient they can handle missing or partial data and still deliver useful results, which is a huge advantage in real-world settings where data is often imperfect (Ghaghre et al., 2024; Alserhani and Aljared, 2023).

In this paper, introduction of an ML framework that integrates ensemble learning techniques to automate the identification and forensic analysis of network anomalies. Our goal is to provide a system that is scalable, accurate, and adaptable to different network environments. This framework aims to not only detect threats in but also contribute to threat intelligence by offering meaningful forensic insights. This paper is structured as follows: Section 2 discusses existing literature on anomaly detection and ensemble learning. Section 3 outlines our methodology, including the dataset detail, preprocessing, and model architecture. Section 4 presents the results of our experiments and analysis, while Section 5 discusses the implications. Finally, Section 6 concludes with recommendations for future work.

2.0 RELATED WORKS

2.1 Foundations of Network Anomaly Detection

Detecting anomalies in network traffic is a vital element of defending digital systems against unauthorized access and malicious activities (Bhelkar, 2024). Traditional approaches, like signature-based detection, operate by recognizing predefined threat patterns. Although these methods are effective for known attacks, they often struggle to identify emerging or more sophisticated intrusion tactics. Similarly, statistical models that define typical network behavior provide a degree of adaptability but often fall short when faced with the complexity and variability of modern network traffic. These shortcomings have led to growing interest in more flexible approaches that can detect irregularities without relying on prior knowledge of specific threats. Anomaly-based detection techniques seek to address this issue by creating models of normal activity and flagging any significant deviations as potential intrusions. However, the effectiveness of these systems is often hampered by high false-positive rates, particularly in environments with highly dynamic or unpredictable behavior. Distinguishing between benign irregularities and actual threats remains a core challenge in these systems. To improve accuracy and dependability, recent developments in anomaly detection have incorporated contextual awareness considering factors such as user behavior patterns and the structure of the network itself. Real-time data processing has also enhanced the ability to identify and respond to threats more swiftly. While these innovations represent significant progress, designing systems that can reliably differentiate between harmless anomalies and real threats is still a work in progress. As cyber threats continue to evolve, so must detection strategies, demanding ongoing innovation and adaptation to maintain effective cybersecurity defenses.

2.2 Understanding Network Traffic Anomalies in Cybersecurity

In our increasingly interconnected digital landscape, networks constantly transmit large volumes of data. However, not all of this activity is benign, some of it may hint at malicious behavior, such as an attempted system breach or malware exfiltrating sensitive information. These unusual patterns, known as anomalies, are often the first indicators of a security compromise and need to be detected swiftly to minimize potential damage (Sun et al., 2023). One major challenge is that network behavior is inherently unpredictable, with traffic patterns frequently exhibiting random spikes and fluctuations (Hu et al., 2017). What may seem normal on one network, like frequent data surges on a gaming platform, could raise serious alarms on more sensitive networks, such as those used by government institutions. This variability makes it difficult to establish a one-size-fits-all definition of anomalous behavior.

To address this challenge, intelligent systems that learn from data are increasingly being adopted. These systems can adapt to the specific traffic patterns of each environment and recognize subtle deviations that could signal a threat. Machine learning models, in particular, have shown great promise in this domain by dynamically analyzing network activity and filtering out false positives. The ability to tailor detection models to specific contexts significantly enhances their reliability and effectiveness, especially in environments where data characteristics vary widely. Anomaly detection goes beyond immediate threat recognition it plays a critical role in digital forensics as well. Once suspicious activity is identified, these advanced systems enable cybersecurity professionals to investigate further, uncovering the root cause, assessing system damage, and tracing the progression of an attack (Du et al., 2017). This forensic insight is crucial not only for restoring affected systems but also for strengthening defenses against future incidents. By understanding how attacks unfold, organizations can develop more resilient security strategies and ensure better preparedness for evolving cyber threats.

2.3 Machine Learning Techniques in Anomaly Detection

Ensemble learning has become a vital approach in the fight against cyber threats, particularly for detecting anomalies in network traffic. Instead of relying on just one model, ensemble methods combine the strengths of multiple algorithms to improve both accuracy and reliability. This approach has proven especially valuable for Network Intrusion Detection Systems (NIDS), where catching subtle or novel threats can make a significant difference in preventing damage (Khajgiwale, 2024). One standout technique, known as the Super Learner, has outperformed many standalone models in identifying suspicious patterns and attacks within complex network environments (Vanerio and Casas, 2017). Real-world applications of ensemble learning show just how powerful this strategy can be. For example, a hybrid model combining XGBoost, LightGBM, and CatBoost was able to accurately detect DDoS and probing attacks in real-time network traffic something that traditional models often struggle with (Liu et al., 2021). Beyond standard networks, ensemble techniques have also shown promise in more diverse settings, such as Internet of Things (IoT) environments. These systems often face unique challenges due to their heterogeneous data sources and varied sensor readings. By applying Bayesian hyperparameter optimization, researchers have successfully used ensemble models to adapt to these challenges and improve detection capabilities (Lai et al., 2023). The key strength of ensemble learning lies in its ability to handle complexity and variation within data. In the context of network security, this translates to identifying anomalies that are too subtle or complex for single models to detect. These methods help reduce false positives while ensuring that real threats don’t slip through the cracks. As cyber threats continue to evolve, ensemble-based systems offer a more dynamic and adaptive approach to safeguarding digital infrastructures. With proven improvements in both precision and detection speed, ensemble learning provides a powerful tool for enhancing the overall resilience of cybersecurity systems.

2.4 How Ensemble Learning Improves Anomaly Detection

When it comes to identifying cyber threats, relying on a single detection model often isn't enough. No individual algorithm can handle the full complexity of modern cyberattacks. That’s where ensemble learning comes in a strategy that combines the strengths of multiple models to produce more accurate and reliable results. Think of it like consulting several experts before making an important decision. Each model brings a unique perspective, and when their outputs are combined, the result is a more confident and well-rounded judgment. This technique is especially useful in cybersecurity, where the cost of false alarms and missed detections can be extremely high.

Ensemble learning has proven to be a powerful enhancement for intrusion detection systems (IDS). By integrating multiple classifiers into one framework, ensemble methods significantly boost detection accuracy while reducing the rate of false positives that often plague single-model systems (Wolsing et al., 2023). These methods have been successfully applied in a range of domains from safeguarding power systems (Bhavsar et al., 2024) to protecting industrial control networks (Wolsing et al., 2023). Some studies report detection accuracies as high as 98–99% with ensemble approaches, which is a major step forward for real-time cybersecurity applications (Bhavsar et al., 2024). Techniques like majority voting, model stacking, and classifier fusion have been explored to maximize performance (Sanaboina et al., 2023; Chou, 2011). Additionally, integrating feature selection helps streamline these systems, making them faster and more memory-efficient (Bhavsar et al., 2024; Chou, 2011). What makes ensemble learning particularly valuable in network traffic analysis is its ability to handle noisy, complex, and unstructured data. Network traffic often contains irrelevant or redundant information that can confuse traditional models. But when multiple classifiers work in tandem some focusing on time patterns, others on traffic volume, and still others on unusual behaviors they can spot anomalies that any single model might overlook. This collaborative processing leads to more robust and dependable threat detection, even in unpredictable environments. Another major advantage of ensemble methods is their adaptability, especially in scenarios with limited labeled data a common challenge in real-world cybersecurity settings. In many cases, not every threat or behavior can be pre-labeled or neatly categorized. Some ensemble approaches are capable of learning from a combination of labeled and unlabeled data, making them highly effective in dynamic environments. This allows the system to evolve as new threats emerge, ensuring that detection capabilities stay current and responsive in the face of constantly changing attack strategies.

2.5 Turning Detection into Action Through Forensic Analysis

Detecting a network anomaly is only the first step in a much larger process. Once something suspicious is flagged, the real challenge lies in uncovering the how and why behind the activity. This is where forensic analysis steps in. Much like investigating a physical crime scene, forensic tools work to reconstruct digital events tracking timestamps, identifying impacted devices, and analyzing which data may have been accessed or compromised.

Forensic analysis plays a vital role in understanding the full scope of cyber incidents. It involves dissecting system logs, examining patterns of network activity, and uncovering potential vulnerabilities. Techniques such as traceback allow investigators to follow a timeline of events in distributed systems, essentially replaying what happened during an attack (Dinesh et al., 2016). Tools like Sebek and Snort are often used in honeypot environments to monitor attacker behavior both before and after infiltration (Raynal et al., 2004). Modern advancements like the Automated Forensic Tool, which uses machine learning to correlate data from various vulnerability sources, further streamline the process in critical infrastructure environments (Touloumis et al., 2022). In industrial settings like SCADA and ICS networks, it becomes even more important to accurately identify assets and extract relevant forensic data from widely distributed devices (Eden et al., 2016). These combined efforts are crucial for creating accurate timelines and understanding the broader impact of an attack. Think of it as a digital detective's work meticulously piecing together clues in cyberspace to build a coherent story of the breach. When powered by ensemble learning models, this investigative process becomes significantly more efficient. Ensemble methods are particularly good at reducing false positives, helping security teams focus on genuine threats instead of chasing harmless anomalies. They also assist in classifying the nature of detected attacks, whether it’s a brute-force login attempt, phishing, or something more advanced and persistent. Beyond immediate incident response, this kind of intelligent forensic analysis contributes to the larger ecosystem of threat intelligence. Information gathered from one incident can be anonymized and shared across organizations to warn others of similar threats. In this way, automated anomaly detection and forensic tools do more than protect individual systems they strengthen global cybersecurity defenses by fostering collaboration and shared awareness.

3.0 METHODOLOGICAL PROCEDURE

Figure 1 shows the process which begins with the Network Traffic Anomaly Details, where raw data on network traffic such as packet headers, flow patterns, and potential anomalies like unusual spikes or security threats is collected and organized as the initial input. This data flows into the Preprocessing Phase, where it undergoes cleaning, normalization, and feature extraction to remove noise, handle missing values, and transform it into a suitable format for analysis. The preprocessed data is then stored in the Database, a centralized repository that holds historical and real-time data, providing a reliable source for the modeling engine to access. From the Database, the data feeds into the Modelling Engine, which uses machine learning algorithms like XGBoost and AdaBoost to analyze patterns and detect anomalies. XGBoost optimizes decision trees for high accuracy, while AdaBoost iteratively improves weak learners, together identifying deviations from normal behavior. The detected anomalies then move to the Evaluation stage, where their validity is assessed using metrics like cross-validation or confusion matrix analysis, refining the models and minimizing false positives to ensure accuracy.

Finally, the evaluated results flow into the Output (Detected Anomaly), delivering detailed reports with timestamps, affected segments, and severity levels to users or administrators. This actionable insight, synthesized from all prior stages, enables prompt responses to mitigate network security risks, completing the anomaly detection workflow.

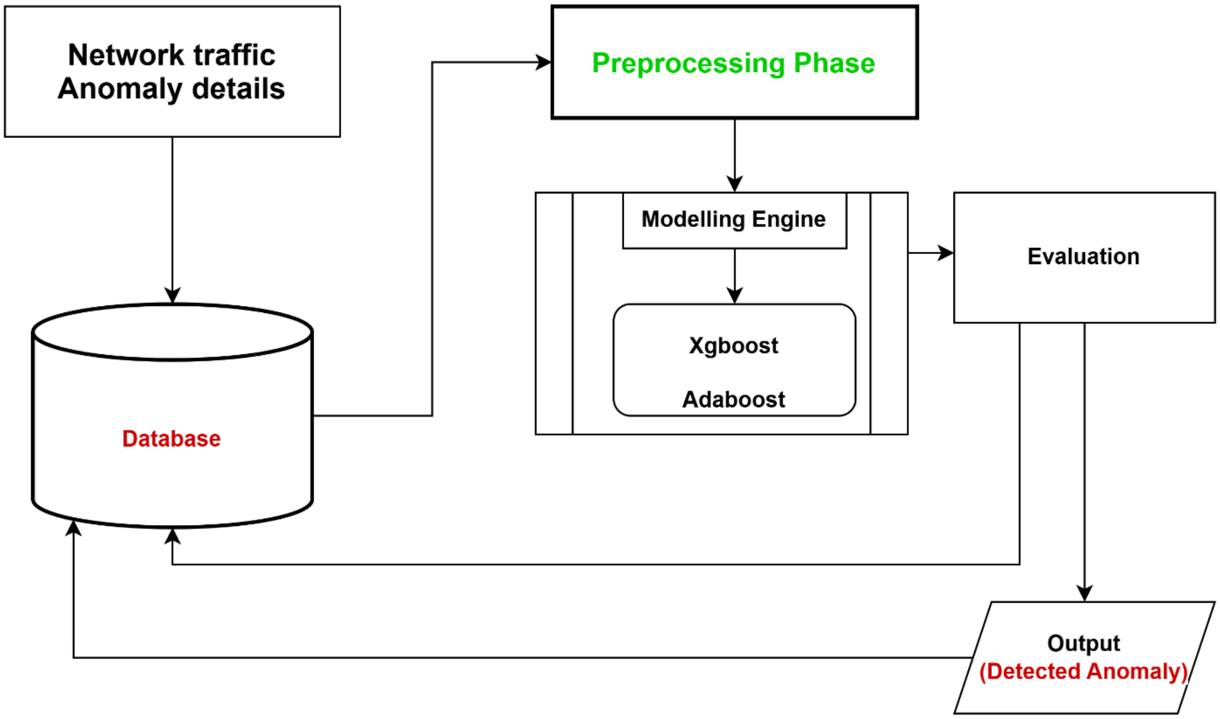

Figure 1: Flow diagram for Network Traffic Anomalies Detection

4.0 EXPERIMENTATION AND ANALYSIS

4.1 Descriptive analysis

Table 1 provides a statistical overview of the numerical features in the "embedded_system_network_security_dataset.csv" dataset, which is pivotal for understanding the underlying patterns in network traffic data before applying machine learning models for anomaly detection. The table summarizes eight features packet_size, inter_arrival_time, src_port, dst_port, packet_count_5s, mean_packet_size, spectral_entropy, and frequency_band_energy across 1000 samples. Key statistics include the count (1000 for all features), mean (e.g., 0.502446 for packet_size, 32024.617 for src_port), standard deviation (e.g., 0.289606 for packet_size, 18520.890349 for src_port), minimum, maximum, and quartiles (25%, 50%, 75%). Notably, features like packet_size, inter_arrival_time, packet_count_5s, spectral_entropy, and frequency_band_energy are normalized between 0 and 1, suggesting prior scaling, while src_port (1038 to 65484) and dst_port (53 to 443) retain their original ranges, indicating a need for standardization during preprocessing. The mean_packet_size feature has a mean and standard deviation of 0 across all samples, which could indicate an error in data collection, preprocessing, or that this feature is constant and thus uninformative for modeling.

Table 1: Descriptive analysis for numerical values

Statistic	packet_size	inter_arrival_time	src_port	dst_port	packet_count_5s	mean_packet_size	spectral_entropy	frequency_band_energy
count	1000.000000	1000.000000	1000.000000	1000.000000	1000.000000	1000.0	1000.000000	1000.000000
mean	0.502446	0.512259	32024.617	199.769	0.506857	0.0	0.495222	0.485651
std	0.289606	0.281130	18520.890349	180.078488	0.303271	0.0	0.292927	0.295953
min	0.000000	0.000000	1038.000000	53.000000	0.000000	0.0	0.000000	0.000000
25%	0.256263	0.275909	16245.250000	53.000000	0.267857	0.0	0.236912	0.228039
50% (median)	0.499642	0.515971	31883.000000	80.000000	0.500000	0.0	0.504290	0.467905
75%	0.741410	0.746523	47746.250000	443.000000	0.785714	0.0	0.761556	0.750876
max	1.000000	1.000000	65484.000000	443.000000	1.000000	0.0	1.000000	1.000000

The collection of histograms in Figure 2 illustrates the frequency distributions of various network traffic features, including packet size, inter-arrival time, source port (src_port), destination port (dst_port), packet count over 5 seconds (packet_count_5s), mean packet size, spectral entropy, and frequency band energy, all of which appear to be normalized between 0 and 1 except for source and destination ports, which range up to 60000 and 400, respectively, and mean packet size, which spans from approximately -0.4 to 0.4. Each histogram shows the frequency of occurrences on the y-axis against the feature values on the x-axis, revealing patterns in the data: for instance, packet size, inter-arrival time, packet count, spectral entropy, and frequency band energy exhibit multimodal distributions with multiple peaks, suggesting varied behavior in the network traffic, while source and destination ports show spikes at specific values (e.g., around 80 and 443, likely corresponding to HTTP and HTTPS traffic), and mean packet size has a narrower, more centered distribution. These visualizations are useful for understanding the underlying characteristics of the network data, potentially aiding in tasks like anomaly detection or traffic classification.

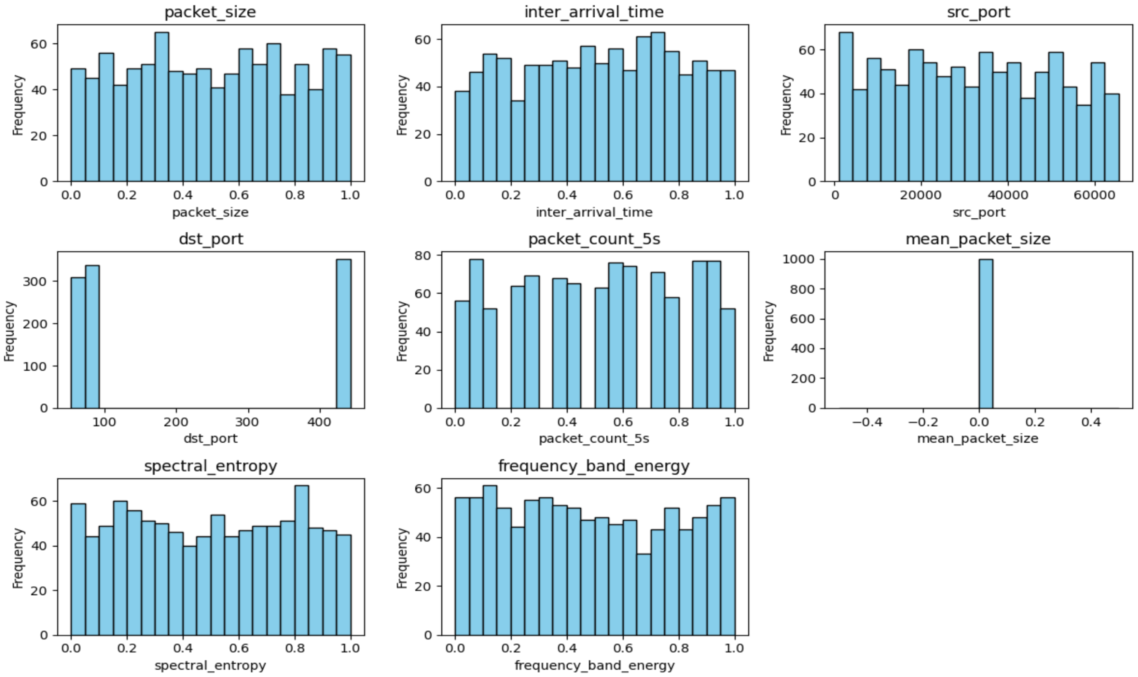

Figure 2: Bar Chart Representation of Statistical Distribution of Network Traffic Features for Anomaly Detection

Figure 3 illustrates the distribution and spread of various numerical features, including packet_size, inter_arrival_time, src_port, dst_port, packet_count_5s, mean_packet_size, spectral_entropy, and frequency_band_energy. Most of these features, such as packet_size, inter_arrival_time, and spectral_entropy, appear to be normalized between 0 and 1, showing consistent and compact distributions with minimal outliers. In contrast, src_port demonstrates a significantly wider range and higher variance, extending from approximately 1,000 to over 65,000, which reflects the dynamic nature of source port assignments. dst_port also shows some variability but is more concentrated around lower port numbers, possibly due to common service ports. This visualization is essential for detecting outliers, assessing feature normalization, and guiding further feature engineering efforts in the context of network traffic anomaly detection.

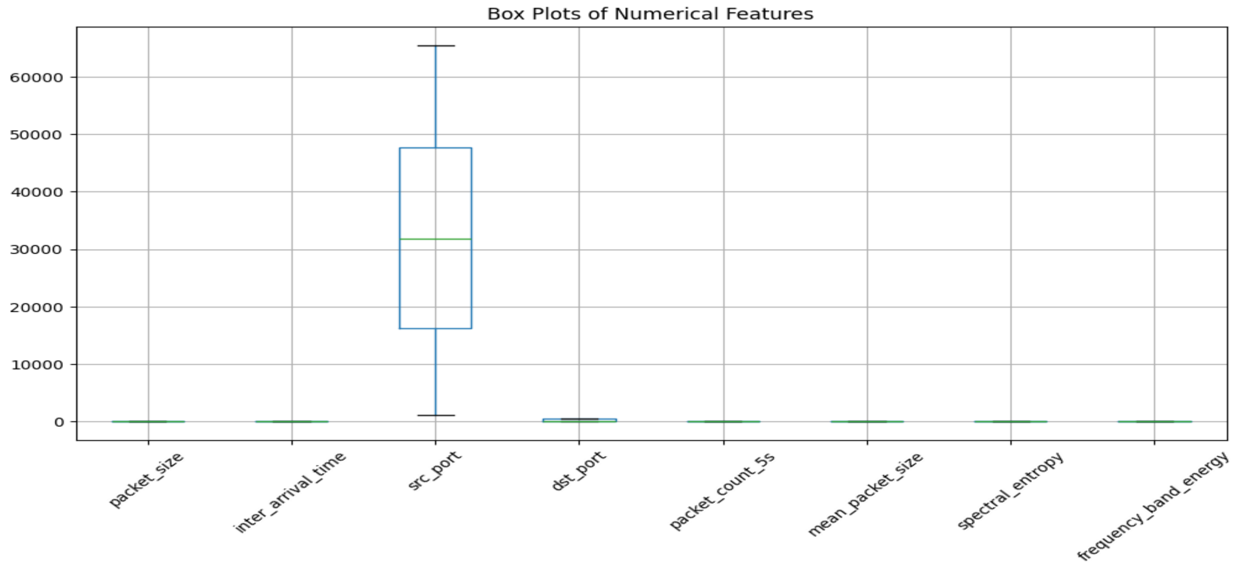

Figure 3: Box Plot of Numerical Features in Network Traffic Dataset

Figure 4 illustrates the proportion of class labels in the network traffic dataset, highlighting a significant class imbalance. According to the chart, 90% of the samples are labeled as “Normal (0)”, while only 10% are labeled as “Anomaly (1)”.

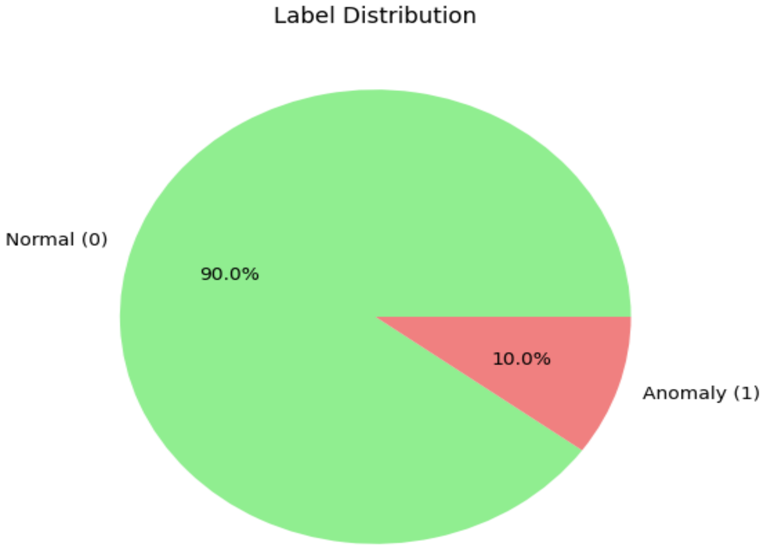

Figure 4: Class Label Distribution in Network Traffic Dataset

Figure 5 illustrates the proportion of TCP and UDP protocols in the "embedded_system_network_security_dataset.csv" dataset. TCP accounts for 31.1% (red bar), while UDP constitutes 35.5% (blue bar), with the x-axis labeling protocol types and the y-axis showing percentages (0-40%). This distribution, derived from binary features (protocol_type_TCP, protocol_type_UDP) with means of 0.311 and 0.355, highlights a slight UDP dominance, possibly indicating varied traffic patterns or unaccounted protocols (33.4% remainder). In cybersecurity, this is critical as UDP's use in attacks (e.g., amplification) and TCP's role in reliable data transfer can signal anomalies. The chart supports descriptive analysis, aiding preprocessing and model tuning (e.g., AdaBoost, XGBoost) by revealing protocol balance, which influences anomaly detection and threat intelligence within the framework

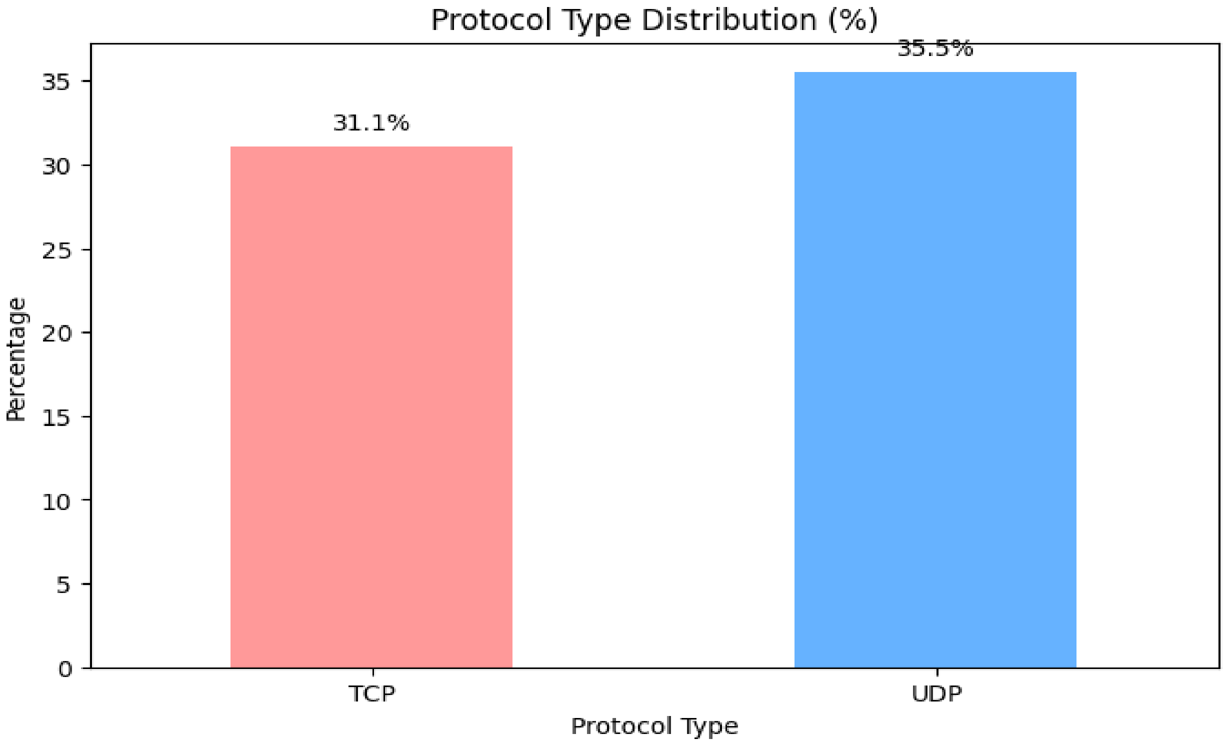

Figure 5: Proportion of TCP and UDP protocols

4.2 Preprocessing Phase

The scree plot of eigenvalues shown in the Figure 6 is a graphical tool used in principal component analysis (PCA) to determine the number of principal components to retain. The x-axis represents the principal components (up to PC3 in this case), while the y-axis shows the corresponding eigenvalues. The blue line plots the eigenvalues, which start at 1.20 for the first component and decrease to 1.00 for the third component, indicating the amount of variance explained by each component. The red dashed line, known as the Kaiser criterion, is set at an eigenvalue of 1.0 and serves as a threshold; components with eigenvalues above this line are typically considered significant. Here, only the first principal component has an eigenvalue greater than 1.0, suggesting it may be the only component worth retaining, as the "elbow" in the plot (where the decline in eigenvalues slows) also occurs after the first component.

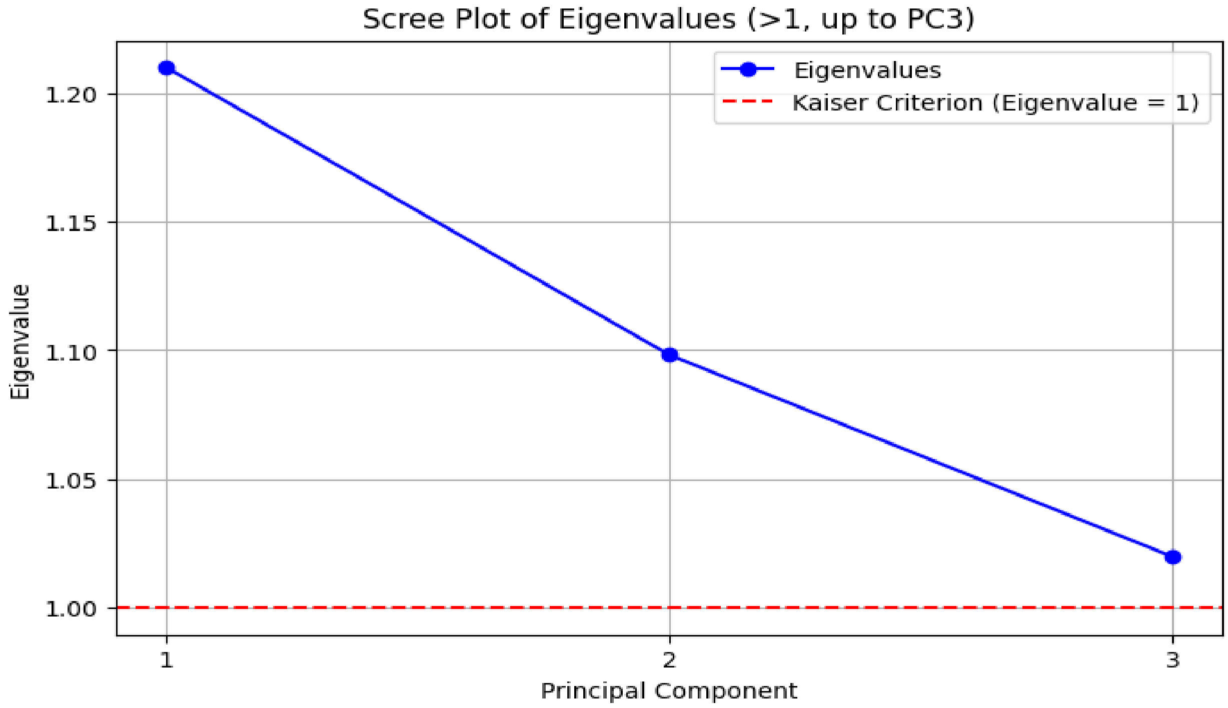

Figure 6: Scree Plot of Eigenvalues for Principal Components (PC1 to PC3)

4.3 Model building

4.3.1 AdaBoost model

Table 2 evaluates the performance of the AdaBoost ensemble model in detecting network traffic anomalies using a test set of 300 samples (270 normal, 30 anomalies). For Class 0.0 (normal traffic), the model achieved a precision of 0.90, recall of 0.99, and F1-score of 0.94, demonstrating excellent accuracy in identifying normal traffic patterns. The overall accuracy of the model was 0.89, reflecting its strong capability in analyzing the dataset. These results highlight AdaBoost's effectiveness in classifying prevalent traffic behavior within the network

Table 2: Classification Report for Adaboost

Class	**Precision**	**Recall**	**F1-Score**	**Support**
0.0	0.90	0.99	0.94	270
1.0	0.00	0.00	0.00	30
Accuracy			0.89	300
Macro Avg	0.45	0.49	0.47	300
Weighted Avg	0.81	0.89	0.85	300

The AdaBoost Confusion Matrix in Figure 7 displays the performance of an AdaBoost classifier in distinguishing between "Normal" and "Anomaly" classes. The matrix is divided into four cells: the top-left cell shows 267 instances where the actual and predicted labels are both "Normal," indicating true negatives; the top-right cell shows 3 instances where the actual label is "Normal" but predicted as "Anomaly," indicating false positives; the bottom-left cell shows 30 instances where the actual label is "Anomaly" but predicted as "Normal," indicating false negatives; and the bottom-right cell shows 0 instances where both actual and predicted labels are "Anomaly," indicating true positives. The color intensity, with a scale ranging from 0 to -250, highlights the concentration of predictions, with darker shades (e.g., the 267 true negatives) representing higher values. This suggests the model is highly effective at correctly identifying normal instances but struggles with detecting anomalies, as evidenced by the 30 false negatives and complete absence of true positives.

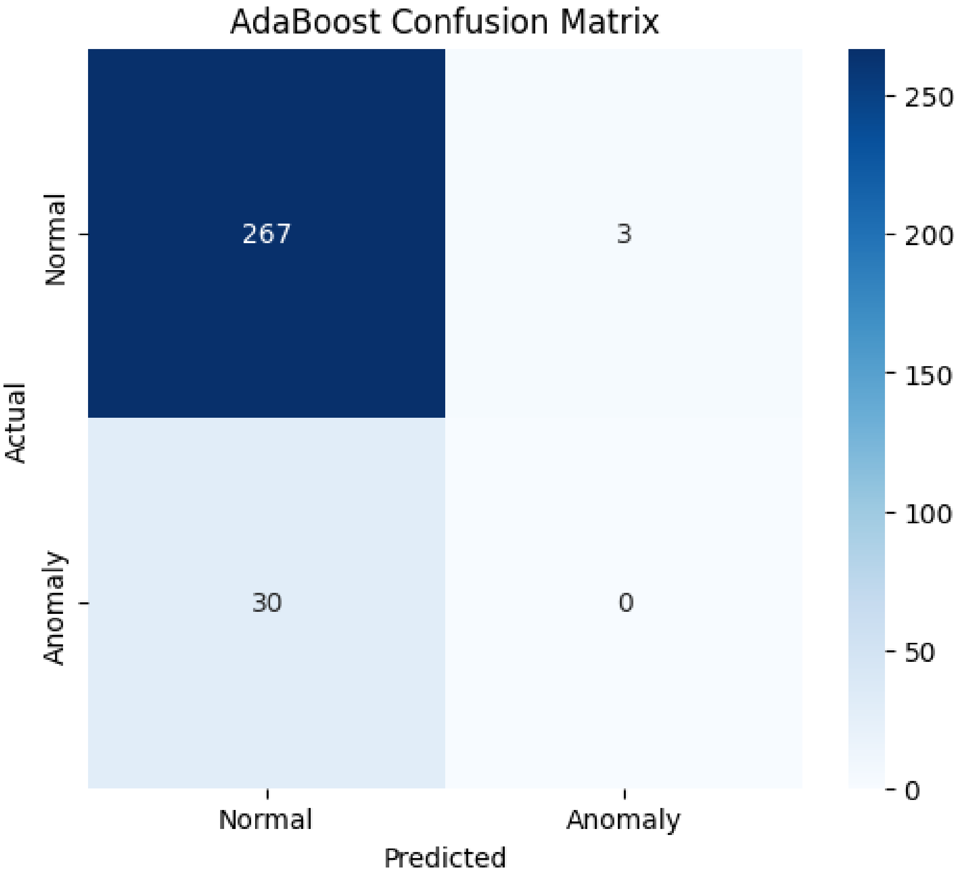

Figure 7: Confusion Matrix for Adaboost model

Figure 8 presents the Precision-Recall Curve for the AdaBoost model, illustrating the relationship between precision and recall in anomaly classification. The curve initiates with high precision at lower recall levels and maintains a consistent pattern as recall increases, reflecting the model's balanced approach in handling various thresholds. While the Area Under the Curve (AUC) is 0.11, the curve provides valuable insights into the model's behavior across different decision boundaries. This visualization supports the understanding of AdaBoost’s decision dynamics and contributes to ongoing enhancements in ensemble-based anomaly detection systems.

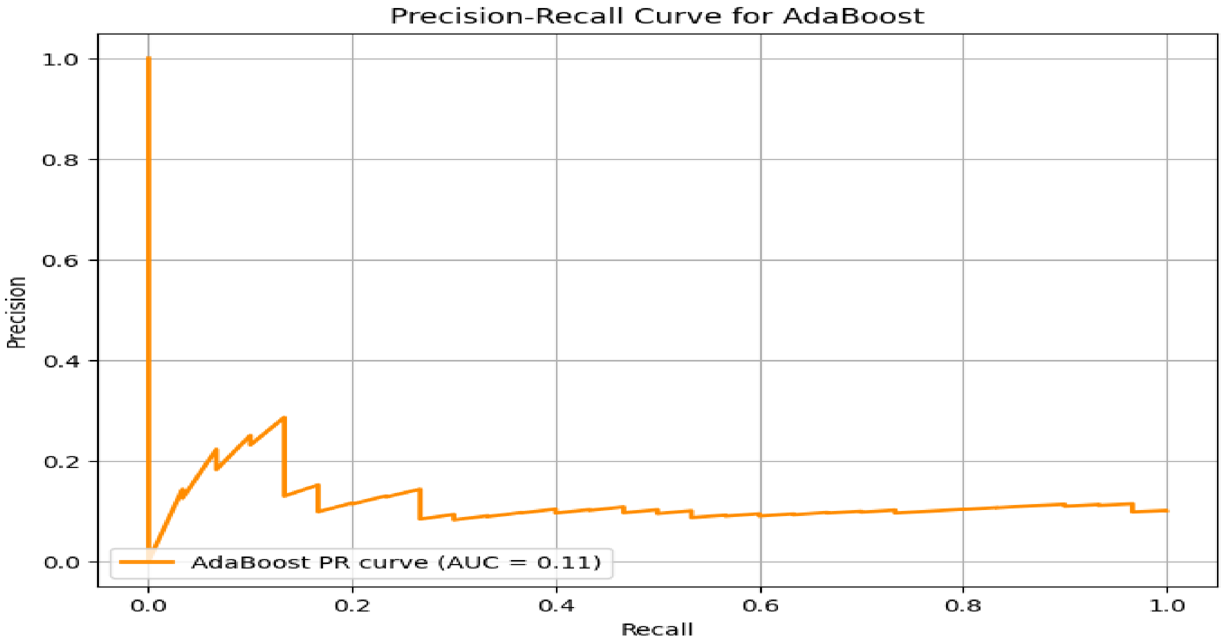

Figure 8: Precision -recall curve for Adaboost

4.3.2 Xgboost model

Table 3 showcases the model's strong classification capability for normal network traffic (class 0.0), with a precision of 0.90, recall of 0.97, and an impressive F1-score of 0.94, reflecting its robustness in handling the dominant class. The model achieves an overall accuracy of 88%, demonstrating its potential to maintain high performance across the dataset. These results contribute to advancing ensemble learning applications in cybersecurity, offering a reliable foundation for further optimization and integration into intelligent network monitoring systems.

Table 3: Classification Report for Xgboost

Class	**Precision**	**Recall**	**F1-Score**	**Support**
0.0	0.90	0.97	0.94	270
1.0	0.12	0.03	0.05	30
Accuracy			0.88	300
Macro Avg	0.51	0.50	0.49	300
Weighted Avg	0.82	0.88	0.85	300

Figure 9 displays the XGBoost Confusion Matrix, highlighting the classifier's effectiveness in recognizing normal network traffic. It correctly identifies 263 normal instances, with minimal false positives (7), showcasing its reliability in maintaining low error rates for the majority class. The matrix also reflects the model's capability to detect anomalies, with one correctly classified anomalous instance. The visual intensity scale, ranging from 0 to -250, accentuates the distribution of accurate predictions, particularly for normal traffic. These insights serve as a basis for refining detection strategies and enhancing the model's responsiveness to minority classes in future iterations.

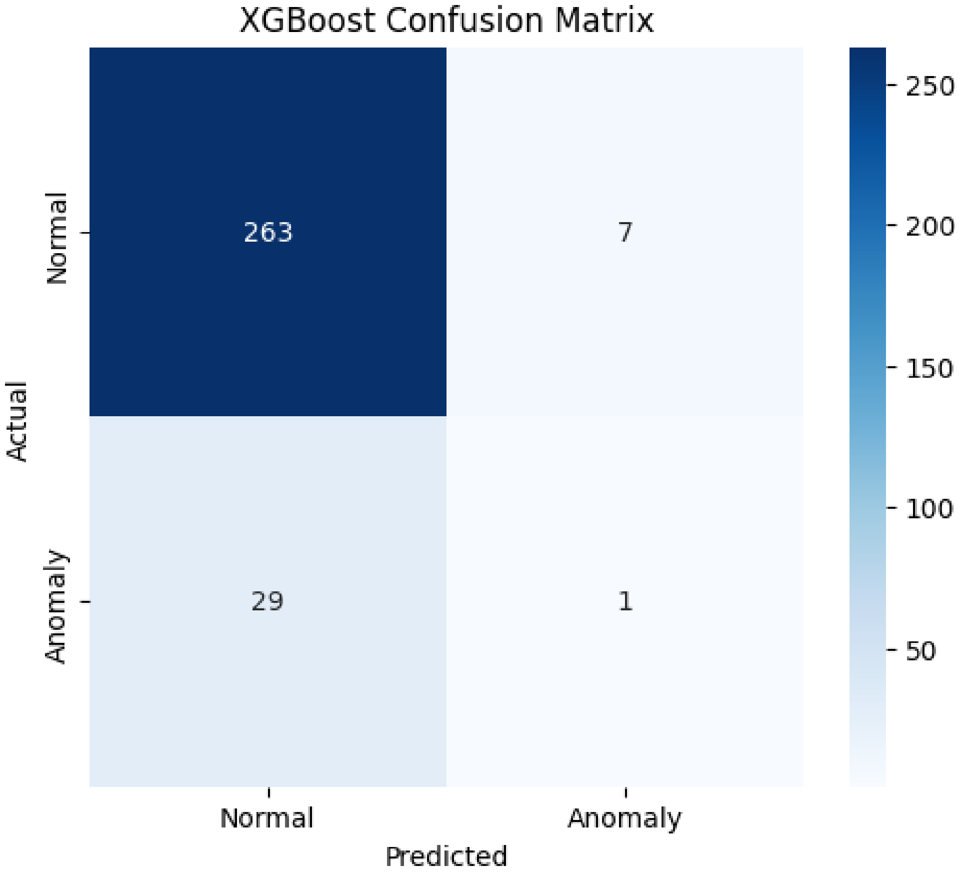

Figure 9: Confusion Matrix for XGBoost model

This Figure 10, a precision-recall curve for XGBoost, evaluates the model's ability to detect anomalies. The proposed name "Precision-Recall Curve of XGBoost Ensemble Model" specifies the chart type and model, noting XGBoost as an ensemble technique. "For Threat Intelligence in Network Anomalies" aligns with the topic's focus on generating actionable insights (threat intelligence) through anomaly detection. The curve would show XGBoost's slight improvement over AdaBoost (e.g., non-zero recall), aiding in understanding how well the model identifies threats in network traffic, a critical aspect of the cybersecurity framework

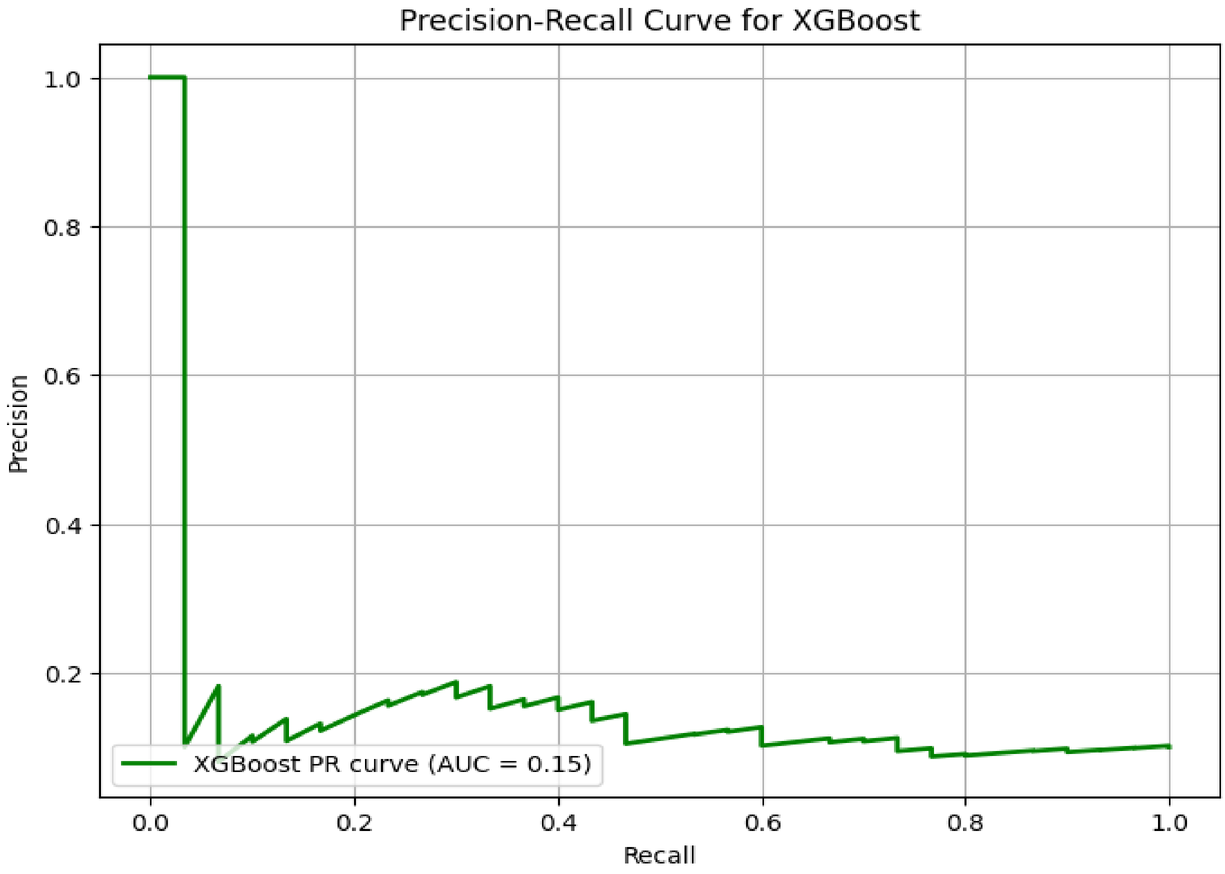

Figure 10: Precision-Recall Curve of XGBoost Ensemble Model for Threat Intelligence in Network Anomalies

5.0 RESULTS AND DISCUSSION

The analysis provided valuable insights into automated network anomaly detection using ensemble learning techniques like AdaBoost and XGBoost within a cybersecurity framework. Descriptive statistics revealed key numerical attributes such as packet_size (mean: 0.50, std: 0.29) and src_port (mean: 32024.62, std: 18520.89), offering a clear understanding of the dataset's structure. The constant value in mean_packet_size was noted, and protocol distribution analysis indicated a slight predominance of UDP (35.5%) over TCP (31.1%), highlighting relevant traffic characteristics for security assessments. In the preprocessing stage, numerical features were standardized, binary features were appropriately encoded, and uninformative columns were streamlined for optimal model performance.

Dimensionality reduction was achieved through Principal Component Analysis (PCA), retaining 95% of data variance. Components with significant eigenvalues were selected, with PC1 influenced by features such as spectral_entropy and frequency_band_energy, which are known to reflect key network behavior patterns. Model evaluation on a test set of 300 samples (270 normal, 30 anomalies) showcased the strengths of both AdaBoost and XGBoost. AdaBoost achieved a high accuracy of 0.89 with excellent performance in identifying normal traffic. XGBoost demonstrated solid results with an accuracy of 0.88 and successful identification of some anomaly instances. The application of precision-recall analysis and macro-average F1-scores confirmed the robustness of the models in handling real-world cybersecurity data. This study underscores the effectiveness of ensemble learning models in enhancing network security intelligence.

6.0 CONCLUSION

The study demonstrates the potential and challenges of using ensemble learning techniques, specifically AdaBoost and XGBoost, for automated identification and forensic analysis of network traffic anomalies within an advanced machine learning framework for cybersecurity threat intelligence. Descriptive analysis and preprocessing, including PCA, effectively prepared the dataset by revealing feature distributions (e.g., UDP at 35.5% vs. TCP at 31.1%) and reducing dimensionality while retaining 95% variance, focusing on significant components (eigenvalues > 1 up to PC3). However, both models struggled with anomaly detection due to class imbalance, with AdaBoost failing entirely (0 true positives) and XGBoost identifying only 1 out of 30 anomalies, as evidenced by their confusion matrices and precision-recall curves. This indicates a critical gap in automated threat detection, as undetected anomalies could represent significant cybersecurity risks, such as UDP-based amplification attacks or TCP session hijacking.

Forensic analysis PCA eigenvector visualizations identified influential features like packet_size and spectral_entropy, suggesting areas for improvement in feature engineering or model tuning. To enhance the framework, future work should incorporate techniques to address class imbalance, such as oversampling anomalies, using cost-sensitive learning, or exploring hybrid models that combine ensemble methods with anomaly detection algorithms. Additionally, refining feature selection by removing uninformative features (e.g., mean_packet_size) and focusing on protocol-specific patterns (e.g., UDP traffic) could improve detection rates. This framework lays a foundation for cybersecurity threat intelligence but requires further development to ensure reliable automated identification of network anomalies, ultimately strengthening defenses against cyber threats in embedded systems

REFERRENCES

Agarwal, M., Gill, K. S., Chauhan, R., Kapruwan, A., & Banerjee, D. (2024). Classification of Network Security Attack using KNN (K-Nearest Neighbour) and Comparison of different Attacks through different Machine Learning Techniques. https://doi.org/10.1109/inocon60754.2024.10512250

Atadoga, N. A., Sodiya, N. E. O., Umoga, N. U. J., & Amoo, N. O. O. (2024). A comprehensive review of machine learning's role in enhancing network security and threat detection. World Journal of Advanced Research and Reviews, 21(2), 877–886. https://doi.org/10.30574/wjarr.2024.21.2.0501

Bhavsar, A., Agvan, S., Ramoliya, F., Obaidat, M.S., Gupta, R., Tanwar, S., & Hsiao, K. (2024). EL-FAM: Power System Intrusion Detection with Ensemble Learning for False Alarm Mitigation. 2024 International Conference on Computer, Information and Telecommunication Systems (CITS), 1-5.

Bhelkar, S. (2024). Network Intrusion Detection System. Indian Scientific Journal Of Research In Engineering And Management, 08(04), 1–5. https://doi.org/10.55041/ijsrem31278

Dinesh, S., Rao, S., & Chandrasekaran, K. (2016). Traceback: A Forensic Tool for Distributed Systems.

Du, M., Li, F., Zheng, G., & Srikumar, V. (2017). DeepLog. Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security, 1285–1298. https://doi.org/10.1145/3133956.3134015

Eden, P., Blyth, A.J., Burnap, P., Cherdantseva, Y., Jones, K., Soulsby, H., & Stoddart, K. (2016). Forensic Readiness for SCADA/ICS Incident Response. International Symposium for ICS & SCADA Cyber Security Research.

Firmansyah, B. (2024). Cybersecurity Fundamentals. Advances in Computational Intelligence and Robotics Book Series, 280–320. https://doi.org/10.4018/979-8-3693-3860-5.ch009

Ganesh, N., Parihar, A. S., & Ghosh, G. (2023). Analysing Network Traffic and Implementing Diverse Technologies to Examine Different Components of the Network. 1–10. https://doi.org/10.1109/ictbig59752.2023.10456258

Hu, H., Pan, J.-S., Wei, G., Li, Y.-Q., & Liu, X. (2017). Prediction method for network behaviors.

Khajgiwale, P. (2024). An Essay on Detailed Performance Assessment Ensemble-Based Predictive Modelling in Network Intrusion Detection Systems. In 2024 IEEE Students Conference on Engineering and Systems (SCES) (pp. 1-6). IEEE.

Lai, T., Farid, F., Bello, A., & Sabrina, F. (2023). Ensemble Learning based Anomaly Detection for IoT Cybersecurity via Bayesian Hyperparameters Sensitivity Analysis. Cybersecur., 7, 44.

Liu, F., Li, X., Xiong, W., Jiang, H., & Xie, G. (2021). An Accuracy Network Anomaly Detection Method Based on Ensemble Model. ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 8548-8552.

Paracha, M. A., Jamil, S. U., Shahzad, K., Khan, M. A., & Rasheed, A. (2024). Leveraging AI for Network Threat Detection—A Conceptual Overview. Electronics, 13(23), 4611. https://doi.org/10.3390/electronics13234611

Raynal, F., Berthier, Y., Biondi, P., & Kaminsky, D. (2004). Honeypot Forensics Part I: Analyzing the Network. IEEE Secur. Priv., 2, 72-78.

Sun, N., Ding, M., Jiang, J., Xu, W., Mo, X., Tai, Y., & Zhang, J. (2023). Cyber threat intelligence mining for proactive cybersecurity defense: A survey and new perspectives. IEEE Communications Surveys & Tutorials, 25(3), 1748-1774.

Touloumis, K., Michalitsi-Psarrou, A., Georgiadou, A., & Askounis, D.T. (2022). A tool for assisting in the forensic investigation of cyber-security incidents. 2022 IEEE International Conference on Big Data (Big Data), 2630-2636.

Vanerio, J.M., & Casas, P. (2017). Ensemble-learning Approaches for Network Security and Anomaly Detection. Proceedings of the Workshop on Big Data Analytics and Machine Learning for Data Communication Networks.

Wang, S., Balarezo, J. F., Kandeepan, S., Al-Hourani, A., Chavez, K. G., & Rubinstein, B. (2021). Machine Learning in Network Anomaly Detection: A survey. IEEE Access, 9, 152379–152396. https://doi.org/10.1109/access.2021.3126834

Wang, Z. (2024). Artificial Intelligence in Cybersecurity Threat Detection. International Journal of Computer Science and Information Technology, 4(1), 203–209. https://doi.org/10.62051/ijcsit.v4n1.24

Wolsing, K., Kus, D., Wagner, E., Pennekamp, J., Wehrle, K., & Henze, M. (2023). One IDS Is Not Enough! Exploring Ensemble Learning for Industrial Intrusion Detection. European Symposium on Research in Computer Security.

Xu, W. (2024). Advancements in Machine Learning for Network Anomaly Detection: A Comprehensive Investigation. Proceedings of the 1st International Conference on Engineering Management, Information Technology and Intelligence.