Enhancing Botnet Detection with Machine Learning and Explainable AI: A Step Towards Trustworthy AI Security

Vishva Patel [1], Hitasvi Shukla[2], Aashka Raval[3]
[1,2,3]Department of Computer Engineering, Pandit Deendayal Energy University, India

Abstract

The rapid proliferation of botnets, armies of compromised machines controlled by malicious actors remotely, has played a pivotal role in the increase in cyber-attacks, such as Distributed Denial-of-Service (DDoS) attacks, credential theft, data exfiltration, command-and-control (C2) activity, and automated exploitation of vulnerabilities. Legacy botnet detection methods, founded on signature matching and deep packet inspection (DPI), are rapidly becoming a relic of the past because of the prevalence of encryption schemes like TLS 1.3, DNS-over-HTTPS (DoH), and encrypted VPN tunneling. These encryption mechanisms conceal packet payloads, making traditional network monitoring technology unsuitable for botnet detection. Faced with the challenge, ML-based botnet detection mechanisms have risen to the top. Existing ML-based approaches, however, are marred by two inherent weaknesses: (1) Lack of granularity in detection because most models are based on binary classification, with no distinction of botnet attack variants, and (2) Uninterpretability, where high-performing AI models behave like black-box mechanisms, which limits trust in security automation and leads to high false positives, thereby making threat analysis difficult for security practitioners.

To overcome these challenges, this study proposes an AI-based, multi-class classification botnet detection system for encrypted network traffic that includes Explainable AI (XAI) techniques for improving model explainability and decision transparency. Two datasets, CICIDS-2017 and CTU-NCC, are used in this study, where a systematic data preprocessing step was employed to maximise data quality, feature representation, and model performance. Preprocessing included duplicate record removal, missing and infinite value imputation, categorical feature transformation, and removal of highly correlated and zero-variance features to minimise model bias. Dimensionality reduction was performed using Principal Component Analysis (PCA), lowering features of CICIDS-2017 from 70 to 34 and those of CTU-NCC from 17 to 4 for maximizing computational efficiency. Additionally, to deal with skewed class distributions, Synthetic Minority Over-Sampling Technique (SMOTE) was employed to synthesise minority class samples to offer balanced representation of botnet attack types.

For CICIDS-2017, we used three machine learning algorithms: Random Forest (RF) with cross-validation (0.98 accuracy, 100K samples per class), eXtreme Gradient Boosting (XGB) with Bayesian optimisation (0.997 accuracy, 180K samples per class), and our recently introduced Hybrid K-Nearest Neighbours(KNN) + Random Forest (RF) model, resulting in state-of-the-art accuracy of 0.99 (180K samples per class). The CTU-NCC dataset was divided across three network sensors and processed separately. Random Forest (RF), Decision Tree (DT), and KNN models were trained independently for each sensor, and to enhance performance, ensemble learning methods such as stacking and voting were applied to combine the results from each of the sensors. The resulting accuracies were as follows: (Random Forest Stacking: 99.38%, Random Forest Voting: 99.35%), (Decision Tree Stacking: 99.68%, Decision Tree Voting: 91.65%), and (KNN Stacking: 97.53%, KNN Voting: 97.11%). Explainable AI (XAI) techniques like SHAP (SHapley Additive exPlanations) and LIME (Local Interpretable Model agnostic Explanation) were integrated to provide enhanced interpretability in eXtreme Gradient Boosting and our Hybrid KNN+Random Forest model, which provided explanations for model decisions and enhanced analyst confidence in the system prediction.

Our key contribution is the Hybrid KNN+Random Forest system with 0.99 accuracy and provision of explainability. We illustrate an accurate, scalable, and deployable AI-based solution for botnet attacks. Our experimentation shows that the multi-class classification method greatly assists in botnet attack discrimination, and Explainable AI (XAI) helps enhance clarity and is thus a strong, practical solution in the real case of botnet detection in an encrypted network scenario.

Keywords: Botnet Detection, Encrypted Networks, Ensemble Models, Explainable AI

I. Introduction

1.1 Background: Understanding Botnets and Their Threats

A botnet is a group of infected devices, usually called bots or zombies, that are infected with malware and under remote control by a botmaster or bot herder. These devices can be personal computers, servers, and Internet of Things (IoT) devices such as smart appliances in a smart home and security cameras used without their owners' permission in cyber attacks. Botnets are frequently used for large-scale automated attacks, utilizing thousands or millions of infected devices to enhance their impact. Common threats associated with botnets include Distributed Denial-of-Service (DDoS) attacks, which flood target systems with excessive traffic, brute force attacks, data exfiltration, spam distribution, financial fraud, and credential stuffing.

Modern botnets have become more advanced tools with capabilities of evading detection and takedown. Contrary to static C2C servers that are easily detectable and blockable , today's botnets use peer-to-peer architecture where the bots communicate with each other in a decentralized manner, making mitigation much more challenging. They use encryption-based communication protocols such as TLS 1.3, DNS-over-HTTPS (DoH), and VPN tunneling that render legacy deep-packet inspection and signature-based detection futile [1]. Botmasters use Domain Generation Algorithms (DGAs) that create dynamically new C2C domains that are not blockable by legacy blocking mechanisms. Encryption, decentralization, and dynamic domain rotation have made detection very challenging [1] and one of the remedies is designing AI-based detection frameworks that are capable of detecting botnet traffic even in the presence of encrypted networks

1.2 The Need for Botnet Detection

Botnet detection is the process of identifying infected network devices and disabling malicious outbound traffic. The conventional detection method depends on signature-based inspection and deep packet inspection (DPI), where patterns in network traffic are analyzed for attack signatures. As encryption technologies have become prevalent, DPI-based detection is no longer effective since new encryption algorithms prevent packet payload access. TLS 1.3, DNS-over-HTTPS (DoH), and VPN tunneling encrypt malicious traffic, making conventional network monitoring mechanisms ineffective .[1]

To address these limitations, machine learning (ML) and artificial intelligence (AI)-driven detection systems are gaining popularity. Unlike signature-based systems, ML-based systems examine network metadata, statistical traffic patterns, and behavioral anomalies and can thus detect new attacks and adapt to new botnet variants. Despite their advantages, AI-driven botnet detection frameworks face multiple challenges, necessitating further improvements in attack differentiation and interpretability.

1.3 Challenges in Existing ML-Based Botnet Detection

While ML-based Intrusion Detection Systems (IDS) have greatly improved botnet detection, the existing models face two major issues. The first one is Limited Attack Differentiation Due to Binary Classification, as the majority of ML-based botnet detection platforms classify network traffic as malicious or benign using binary classification models [2]. While adequate for general threat detection, these models cannot identify specific botnet attacks, i.e., DDoS, brute force, data exfiltration, and C2 communication. Such lack of granularity lowers the efficacy of threat prioritization and mitigation. The second major issue is Opaque AI Decision-Making and Lack of Explainability, as most AI-driven botnet detection models operate as black boxes and do not provide sufficient insight into their decision-making process [3]. It is difficult for security analysts to know why a specific network flow is labeled malicious, raising false positive rates and lowering confidence in AI-based security solutions [4].

1.4 Proposed Approach

To address these challenges, we propose a multi-class classification model for botnet detection in encrypted networks, integrating Explainable AI (XAI) techniques for enhanced explainability and trust. We employ two publicly available datasets [1], CICIDS-2017 [5] and CTU-NCC [6], and follow a strict data preprocessing pipeline to ensure high-quality feature representation and optimal model performance. The preprocessing steps include duplicate entry elimination, which removes redundant entries to prevent bias during model training, and handling missing and infinite values, where median imputation is used for missing values and infinite values are handled by assigning NaN (Not a Number). Additionally, encoding categorical features converts non-numeric attributes into machine-learning-suitable representations, and feature correlation analysis identifies and removes strongly correlated features [4] to avoid multicollinearity and overfitting. Furthermore, dimensionality reduction is applied using Principal Component Analysis (PCA), reducing the CICIDS-2017 [5] dataset from 70 to 34 features and the CTU-NCC [10] dataset from 17 to 4 features to enhance computational efficiency [7]. Finally, the class imbalance [4] was handled with the help of SMOTE by generating artificial samples for minority attack classes.

1.5 Machine Learning Models and Performance

To evaluate our approach, we trained multiple ML models on both datasets. For the CICIDS-2017 [5] dataset, Random Forest (RF) with Cross-Validation achieved 0.98 accuracy with 100K samples per class. Similarly, eXtreme Gradient Boosting (XGB) with Bayesian Optimization attained 0.997 accuracy with 180K samples per class. Moreover, the Hybrid KNN+RF model with Cross-Validation recorded the highest accuracy of 0.99 in our experiments with 180K samples per class. To enhance model interpretability, SHAP, and LIME were embedded into the XGB and Hybrid KNN+RF models, providing detailed insights into feature contributions and model decision rationales. For the CTU-NCC [6] dataset, which involved processing data from three network sensors separately, ensemble techniques were used to enhance the performance of each model. The Random Forest stacking and voting models showed accuracy scores of 99.38% and 99.35%, while the Decision Tree recorded 99.68% for stacking and 91.65% for voting. Similarly, KNN achieved 97.53% for stacking and 97.11% for voting.

1.6 Contributions and Impact

This study introduces a new hybrid K-Nearest Neighbors + Random Forest (KNN+RF) model with an unprecedented accuracy of 0.99 while incorporating Explainable AI (XAI) for greater interpretability. Our contributions include a cutting-edge Hybrid KNN+RF model, which achieves a record 0.99 accuracy, beating the existing models for botnet detection. Moreover, we present a multi-class classification framework that, unlike existing binary classification models, discriminates between multiple botnet attack types, enhancing specificity and detection rates. Moreover, by Explainable AI (XAI) integration using SHAP and LIME, our model provides greater transparency and interpretability, allowing security analysts to validate and trust AI-driven threat detections. Ultimately, this paper offers a scalable, high-accuracy, and explainable AI-driven botnet detection system, presenting a deployable real-world security solution for encrypted networked domains.

II. Literature Survey

The AI-powered botnet detection space has seen remarkable progress in recent years. Conventional detection techniques, including signature-based intrusion detection and deep packet inspection (DPI), have become increasingly ineffective with the ubiquitous use of encryption protocols like TLS 1.3, DNS-over-HTTPS (DoH), and VPN tunneling. Since encryption makes packet payload inspection challenging, machine learning-based detection techniques have become more significant by examining network flow patterns and behavioral anomalies [5]. Nevertheless, despite remarkable progress, most critical challenges are still unsolved, affecting model accuracy, interpretability, and real-world usability.

To review existing literature and highlight research gaps, we conducted a detailed review of 20 studies on various AI-based botnet detection methods. The following sections explain the most prevalent methods, their shortcomings, and the research gaps addressed in this work.

2.1 AI-Based Approaches for Botnet Detection

Several studies have investigated different machine learning-based approaches for botnet detection in network traffic. One of the most common approaches is binary classification models, which classify traffic as either benign or malicious [2]. The study in [8] employed Graph Convolutional Networks (GCN) to fuse flow and topology features for botnet detection. Meng, Xiaoyuan, [8] noted that existing botnet detection methods usually only use one kind of features, i.e., flow features or topology features, which overlooks the other type of features and affects the model performance. Meng, Xiaoyuan, [8] proposed constructing communication graphs from network traffic and representing nodes with flow features, achieving a recall rate of 92.90% and an F1-score of 92.76% for C2 botnets. However, this approach primarily differentiates between malicious and non-malicious traffic, failing to identify specific attack types. To address this limitation, multi-class classification models provide a more refined approach by identifying different botnet attack types. The research in [9] proposed stacking Deep Convolutional Neural Networks (CNN), Bi-Directional Long Short-Term Memory (Bi-LSTM), Bi-Directional Gated Recurrent Unit (Bi-GRU), and Recurrent Neural Networks (RNN) for botnet attack detection. Using the UNSW-NB15 [8] dataset, A. K. Kumar et al. [9] reported a testing accuracy of 99.76% and stated that the proposed model accurately provides for the intricate patterns and features of botnet attacks, achieving a high ROC-AUC curve value of 99.18%. However, severe class imbalance in [10] and [4] limits their robustness, leading to poor classification performance on minority attack types. Beyond traditional botnet detection, a critical area of research is the identification of encrypted network traffic, where AI models rely on network flow-based features instead of packet payload inspection due to the prevalence of encrypted communication protocols. The study in [7] proposed a framework for encrypted malicious traffic detection, noting that the popularity of encryption mechanisms poses a great challenge to malicious traffic detection, as traditional detection techniques cannot work without decrypting encrypted traffic. The framework is a two-layer detection system that outperforms classical deep learning and traditional machine learning algorithms, such as ResNet and Random Forest [7]. However, despite its advantages, high false positive rates and low real-world verification hinder its practical application.

2.2 Identified Research Gaps and Contributions

Based on our literature review, we have identified five major research gaps from, [1], [2], [7], [8], and [9]. This research primarily fills in two of the most critical research gaps, which are required to continue botnet detection research. The first is multi-class classification and anomaly detection, where most ML-based botnet detection models are still binary [2] classification-based [8], failing to distinguish between specific botnet attack types such as DDoS, brute force, data exfiltration, and command-and-control (C2) activities. Binary classifiers [8] lack actionable information as they only detect malicious presence and not the type of botnet attacks, hence effective mitigation plans are difficult. To address this, we introduce a multi-class classification model that can identify and classify multiple types of botnet attacks in encrypted traffic. With the inclusion of SMOTE for class balancing [4], the model enhances detection in all attack types, avoiding majority class bias and overall detection performance enhancement. The second major gap is Explainable AI (XAI) for encrypted traffic analysis. AI-based botnet detection models are not explainable, given that they act as black-box classifiers exhibiting low transparency in decision-making. Security practitioners must understand why AI-driven decisions are being taken, especially in mission-critical cybersecurity operations, where a missed threat leads to operational downtime or false positives. Therefore, we applied SHAP and LIME to XGB and Hybrid KNN+RF models to provide transparency to the decision-making process in the models. These two techniques provide insights into feature importance, allowing security analysts to validate and trust AI-driven threat detection with fewer false positives.

2.3 Comparative Analysis of Existing Work

Table I presents a comparative analysis of existing studies [7], [8], [9] on botnet detection, highlighting key aspects such as classification type, dataset, XAI integration, and model accuracy.

TABLE I: COMPARISON OF EXISTING BOTNET DETECTION MODELS

Study	Approach	Dataset	Classification Type	XAI Integration	Best Accuracy	Key Limitation
Deeply Fused GCN	Graph Neural Networks (GCN)	CTU-13	Binary	No	0.96	Missing Multiclass Classification
Feature Mining ML	RF, XGB	CICIDS-2017 [9]	Binary	No	0.98	High false positive rate
Hybrid Deep Learning	LSTM, CNN	Bot-IoT	Multi-Class	No	0.95	Poor classification for minority attack classes
Our Approach	RF, XGB, KNN+RF	CICIDS-2017 [9] & CTU-NCC [10]	Multi-Class	Yes (SHAP & LIME)	0.99	State-of-the-art accuracy with XAI

2.4 Conclusion of Literature Review

Although AI-based botnet detection has seen advancements, significant challenges persist. Most existing models use binary classification, failing to discriminate between individual attack types [2]. Few studies integrate explainable AI, making ML-based cybersecurity models uninterpretable and untrustworthy. This study explicitly addresses two significant research gaps by introducing multi-class classification, which improves accuracy and detection granularity by classifying various botnet attack types, unlike binary classification models. Additionally, we integrate Explainable AI (XAI) for encrypted traffic analysis. Our framework facilitates model interpretability by combining SHAP and LIME, enabling increased trust, transparency, and analyst adoption. While our current setup is offline, the future is toward real-time deployment with minimal inference latency. Adaptive feature selection and federated learning are also areas of future research to make botnet detection adaptive to emerging threats, scalable, and privacy-preserving. Our Hybrid KNN+RF approach (accuracy 0.99) surpasses current approaches and offers a deployable, scalable, and explainable AI-based botnet detection system. This work lays a strong foundation for AI-enabling cybersecurity solutions by bridging the gap between the practical usability of real-world approaches in encrypted networks and high-accuracy detection.

I. Methodology

3.1 Dataset Description and Preprocessing

In this study, we utilize two datasets to detect and classify multiple botnet attack types in encrypted network traffic. The CICIDS dataset is a well-known dataset containing real-world network traffic data with labeled attack and benign instances. It consists of 2.8 million rows and 79 features, covering eight different types of network activity classes. Additionally, we integrate the CTU and NCC datasets to form the CTU-NCC Combined dataset, which provides a diverse range of botnet attack traffic. This dataset contains 14.7 million rows and 18 features, also categorized into eight unique network activity classes. These datasets were selected in order to enable a thorough assessment of multi-class botnet identification in various network configurations.

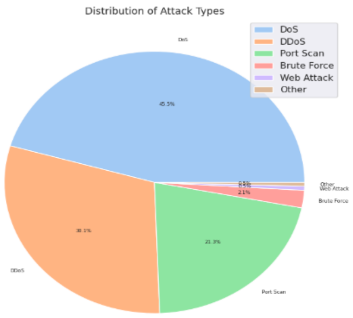


Figure 1: CICIDS Attack Type Distribution

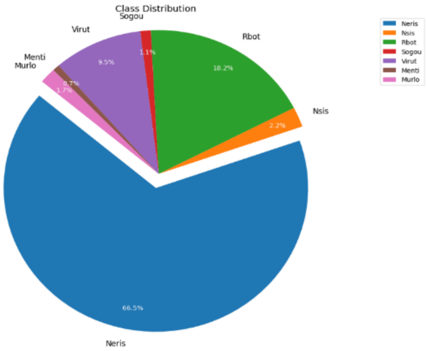


Figure 2: CTU-NCC Attack Type Distribution

The datasets underwent a number of preprocessing and cleaning procedures in order to be ready for classification. Duplicate records were identified and removed to maintain data integrity, while infinite values were replaced with appropriate numerical values to avoid computational errors. Label encoding was used to convert categorical attack labels into numerical values, ensuring compatibility with machine learning models, and additional categorical encoding was applied where necessary. Feature selection was conducted using correlation analysis, retaining attributes that exhibited strong positive correlations with specific botnet attack types to enhance classification performance.

Given the class imbalances in botnet attack types, data sampling techniques were applied to achieve a more balanced distribution between attack and benign instances. Outlier detection and handling were performed to assess their impact on model training, using techniques such as removing extreme outliers or transforming skewed features while preserving attack-specific patterns. Features with zero variance were eliminated to reduce dimensionality and computational overhead.

Standardization using z-score normalization was applied to ensure that all features contributed equally to the model and facilitated faster convergence in machine learning algorithms.

To further optimize the feature space, Principal Component Analysis (PCA) was applied for dimensionality reduction while retaining the most informative components relevant to botnet detection. The number of principal components was selected based on variance retention criteria to achieve an optimal balance between dimensionality reduction and information preservation. Through these preprocessing and transformation steps, we ensured that the datasets were well-prepared for accurate and efficient multi-class botnet detection in encrypted network traffic.

3.2 Model Implementation

To classify network traffic and detect botnet activity, multiple machine learning models were implemented on the CICIDS dataset. The Random Forest Classifier was trained with 15 estimators, a max depth of 8, and 30 max features, and its performance was evaluated using 5-fold cross-validation. The XGBoost Classifier was optimized through Randomized Search with cross-validation, followed by Bayesian Optimization to refine hyperparameters for improved accuracy. Additionally, a Hybrid Model (Voting Classifier) was developed by combining K-Nearest Neighbors (KNN) and Random Forest using soft voting. This model incorporated feature scaling and classification through a pipeline, and its performance was assessed based on accuracy, precision, recall, and F1-score.

For the CTU-NCC Combined dataset, models were first trained separately on three different sensors before being combined into ensemble models using Voting and Stacking. Individual sensor-based models included Random Forest (10 estimators, max depth of 6, no feature limit), Decision Tree (max depth of 8), and KNN (8 neighbors, distance-based weighting with the Manhattan distance metric), each evaluated using 5-fold cross-validation. The ensemble models combined sensor-specific models to enhance classification accuracy. A hard voting classifier was built using the three KNN models trained on separate sensors, while a stacking model used these KNN models as base estimators with Logistic Regression as the final estimator. Similarly, Random Forest and Decision Tree-based ensemble models were developed, following the same voting and stacking approach to leverage patterns from all three sensors. The ensemble models were evaluated on the test set to assess their effectiveness in improving classification performance.

3.3 Explainability AI

To enhance model interpretability, SHAP (SHapley Additive exPlanations) and LIME (Local Interpretable Model-agnostic Explanations) were employed. SHAP was used to analyze global feature importance in both XGBoost and Random Forest models, while local explanations for individual predictions provided insights into model decisions.LIME was used to visualize the influence of particular characteristics on classification results by explaining individual predictions for Random Forest and KNN classifiers. These explainability strategies made sure that the models were transparent, interpretable, and performance-optimized, which made them useful for multi-class botnet detection in encrypted network traffic.

III. Results

4.1 CICIDS Dataset

To compare performance among models for network intrusion detection, we trained and tested four classifiers using the CICIDS dataset for multi-class classification: Random Forest (RF), XGBoost with Random Search Optimization (XGB-RS), XGBoost with Bayesian Optimization (XGB-BO), and a Hybrid Model combining K-Nearest Neighbors (KNN) and Random Forest. Classifiers were compared on accuracy of classification, which revealed the Hybrid model outperformed the rest at 99.9% accuracy, followed by XGB-BO at 99.8%, XGB-RS at 99.7%, and Random Forest at 98.2%. Despite the Random Forest classifier showing robustness, it fell short of the rest due to the lack of iterative learning, fixed hyperparameters, and poor ability to capture complex interactions between features. Contrary to this, XGBoost with Random Search Optimization boosted precision by leveraging boosting algorithms, optimal hyperparameter tuning, and improved management of class imbalance. Additional improvement with Bayesian Optimization provided marginally improved results through optimal choice of hyperparameters, regularization operations, and balance of model complexity. The Hybrid model integrating KNN and Random Forest successfully captured local patterns and global robustness through ensemble voting, achieving an overall highest accuracy level. Through these observations, it was made clear that the power of ensemble learning and boosting approaches to maximizing network intrusion detection ability exists with the Hybrid model being proven to possess the largest generalization ability among types of attacks.

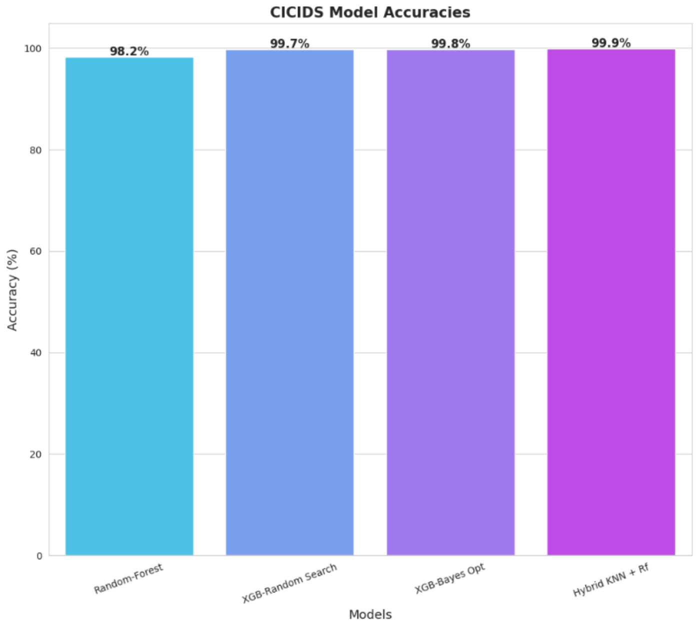


Figure 3: CICIDS Model Accuracies

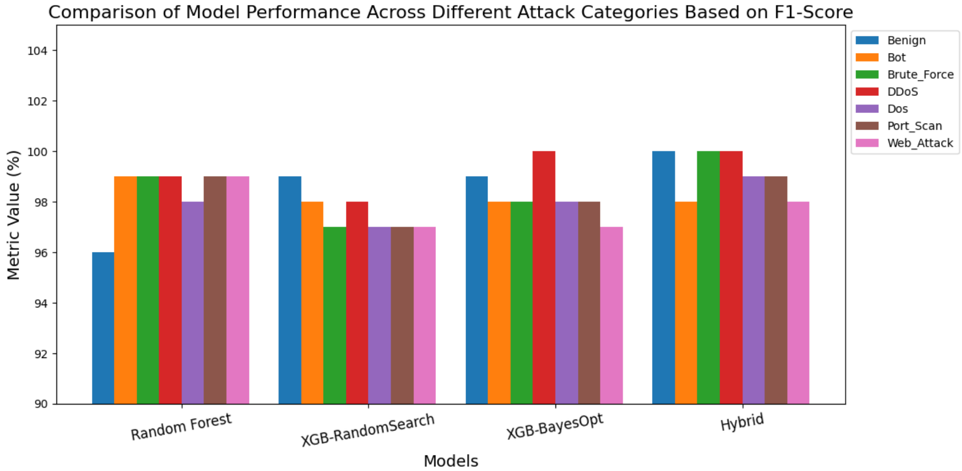


Figure 4: CICIDS Model Performance Across Attack Categories

4.2 CTU-NCC Dataset

To evaluate the effectiveness of different classification models for botnet detection, we implemented six ensemble-based models using Voting and Stacking techniques across three base classifiers: K-Nearest Neighbours (KNN), Random Forest (RF), and Decision Tree (DT). The models were assessed based on their classification accuracy, as shown in Table II. The results indicated that the KNN-based models performed decently by way of accuracy, where the KNN-Voting model achieved 97.11% and the KNN-Stacking model marginally better at 97.53%. This slight improvement suggests that stacking, where the predictions from numerous KNN models are aggregated and combined by a meta-learner, had better generalization when compared to majority voting. However, both KNN models were outperformed by tree-based ensembles due to KNN's sensitivity to data distribution, computational complexity, and lack of inherent feature importance weighting. Random Forest-based models demonstrated high performance, with the RF-Voting model achieving 99.35% accuracy and the RF-Stacking model slightly improving to 99.38%. The minimal difference between these models suggests that while stacking provides refinement, the inherent robustness of Random Forest through bagging, feature importance handling, and variance reduction already ensures strong performance. Decision Tree-based models exhibited the most significant variation, with the DT-Voting model scoring the lowest accuracy at 91.65%, while the DT-Stacking model achieved the highest accuracy of 99.68%. This drastic improvement highlights the vulnerability of individual decision trees to overfitting and the effectiveness of stacking in mitigating this issue by leveraging multiple weak learners and refining predictions through a meta-learner. These findings suggest that stacking generally outperforms voting, with the improvement being most pronounced in Decision Trees. Random Forest models were consistently good, and hence a safe bet for botnet detection, while Decision Trees gained the most from stacking, with a substantial improvement in their predictive power. While KNN models were good (~97%), they were surpassed by tree-based ensembles, probably because they could not learn complex feature interactions well. In general, the DT-Stacking model was the top performer with 99.68% accuracy, roving that a hierarchical stacking framework can be very effective for botnet detection, while RF-based ensembles were strong contenders because of their stability and consistency.

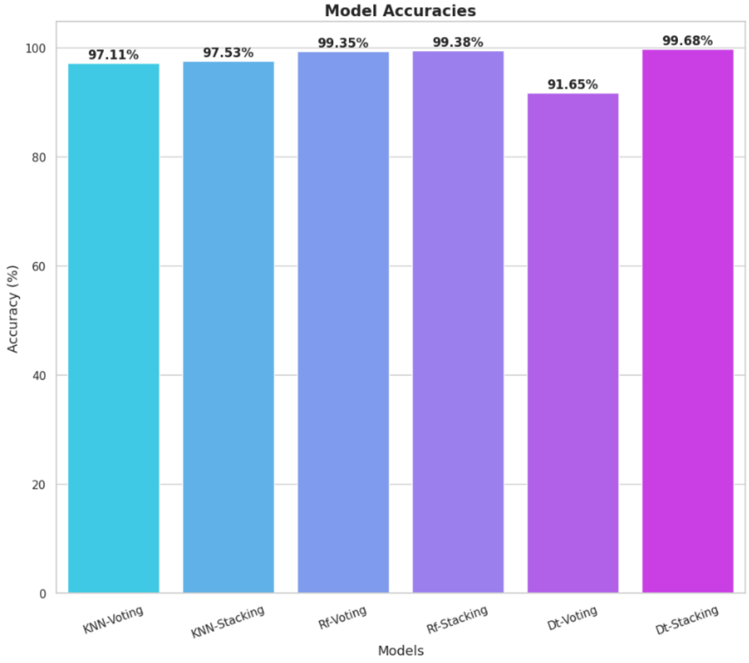


Figure 5: CTU-NCC Model Accuracies

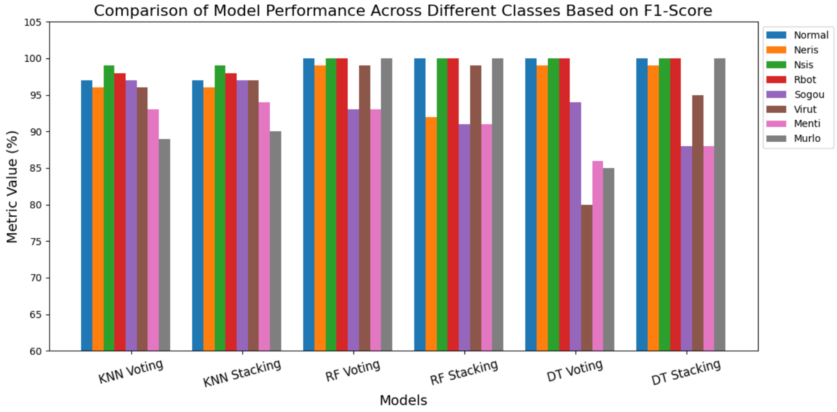
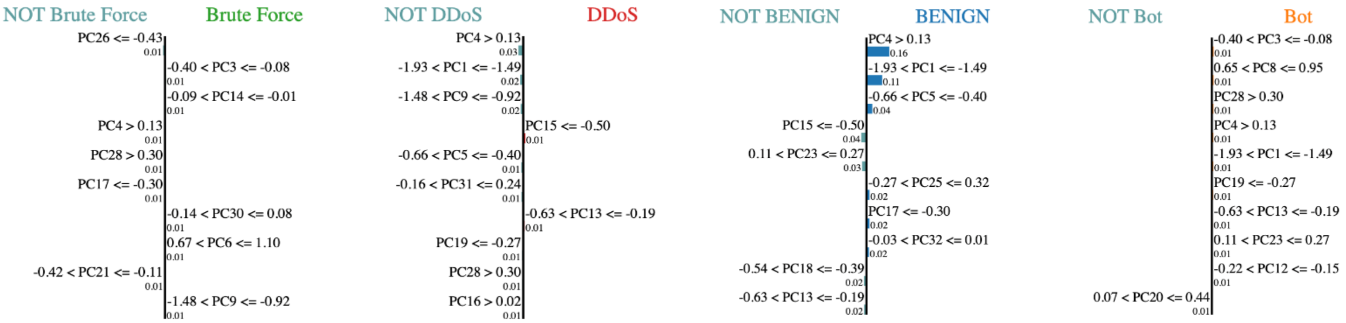


Figure 6: CTU-NCC Model Performance Across Different Attack Classes

4.3 Explainability AI

Understanding machine learning model decision-making is just as important as accuracy, especially in the field of cybersecurity, where transparency and trust are top priorities. Explainable AI (XAI) techniques were used in this research to the CICIDS dataset to examine our intrusion detection model's decision-making process. The primary objective was to learn about the model's reasoning for labeling certain network traffic as an attack and the most influential factors on these predictions. Utilizing XAI tools such as SHAP (Shapley Additive Explanations) and LIME (Local Interpretable Model-Agnostic Explanations), we discovered significant features with a considerable influence on traffic classification. Some of the important findings included the importance of flow duration, as abnormally long network connections were likely to be indicative of suspicious traffic; the number of forward and backward packets, which were good indicators of malicious activity; packet length variance, whose extreme variance was often seen in conjunction with cyber attacks; and the use of source and destination ports, as abnormal port behavior was often found together with a specific type of attack. Visualization of the model's decision-making process using SHAP plots and LIME explanations allowed greater appreciation of how normal traffic and attack traffic were distinguished. Clear patterns were discernible for threats like DDoS and brute force attacks, thereby making them easier to spot, while false positives became easier to understand, allowing us to improve the model by fine-tuning thresholds and optimizing feature selection. The incorporation of XAI not only enhanced the performance of the model by increasing accuracy and reducing false alarms but also allowed greater confidence in its decision-making by making it more transparent and interpretable. By using explainability techniques, we established a connection between effectiveness and trust, thereby making our intrusion detection system characterized by high accuracy and enhanced reliability and interpretability.



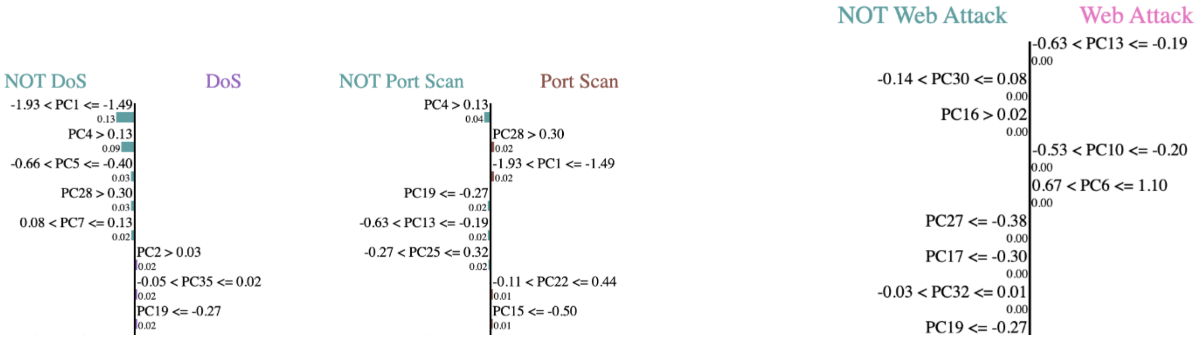
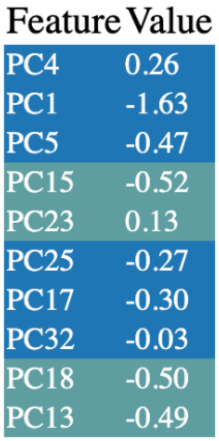
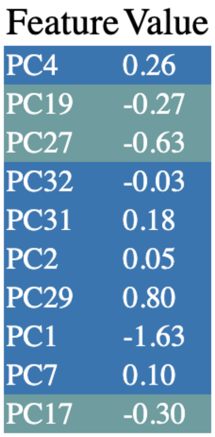


Figure 7: LIME Graphs For KNN



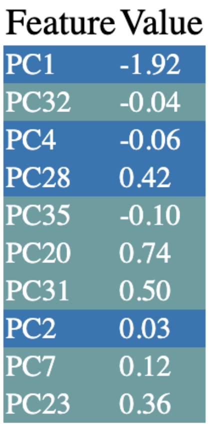
Feature	Value
PC4	0.26
PC1	-1.63
PC5	-0.47
PC15	-0.52
PC23	0.13
PC25	-0.27
PC17	-0.30
PC32	-0.03
PC18	-0.50
PC13	-0.49

Figure 8: Feature Value Graph For KNN



Feature	Value
PC4	0.26
PC19	-0.27
PC27	-0.63
PC32	-0.03
PC31	0.18
PC2	0.05
PC29	0.80
PC1	-1.63
PC7	0.10
PC17	-0.30

Figure 9: Feature Value Graph For Rf



Feature	Value
PC1	-1.92
PC32	-0.04
PC4	-0.06
PC28	0.42
PC35	-0.10
PC20	0.74
PC31	0.50
PC2	0.03
PC7	0.12
PC23	0.36

Figure 10: Feature Value Graph For XGB

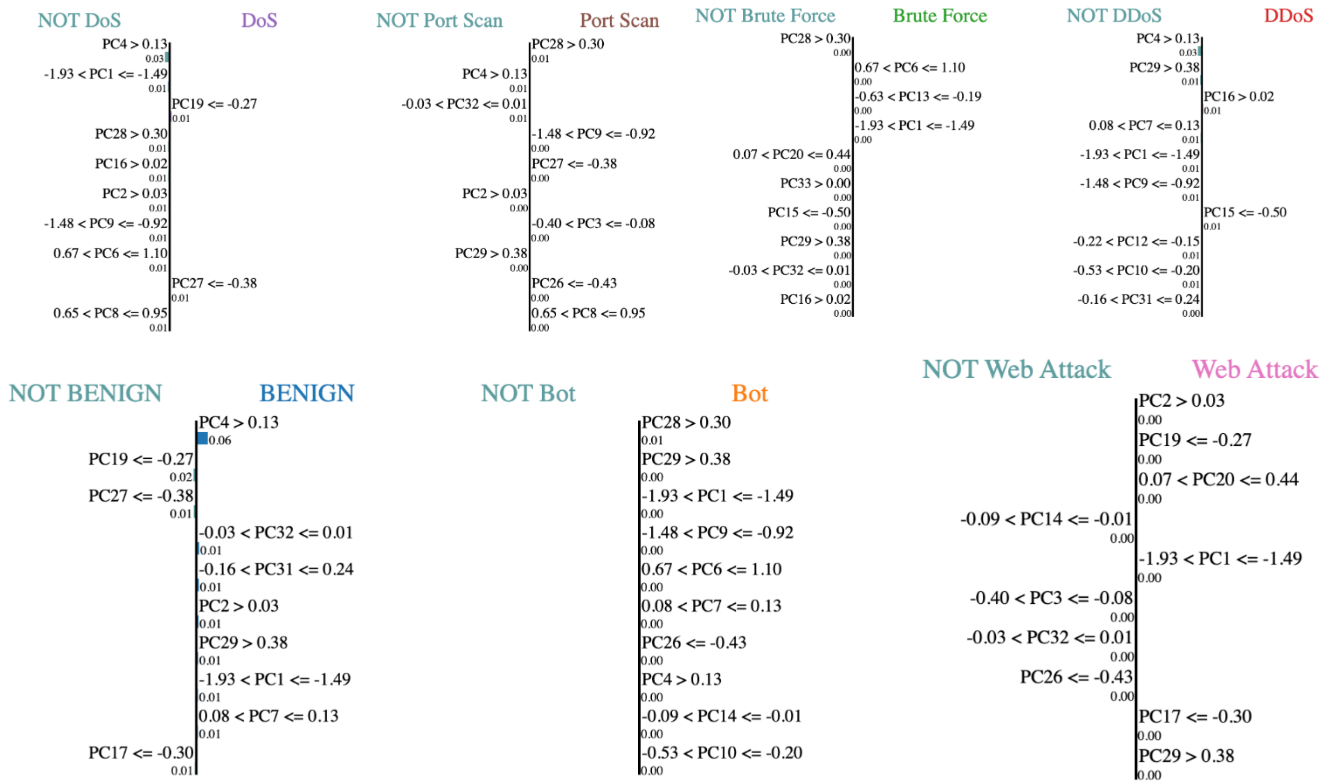


Figure 11: LIME Graphs for Random Forest

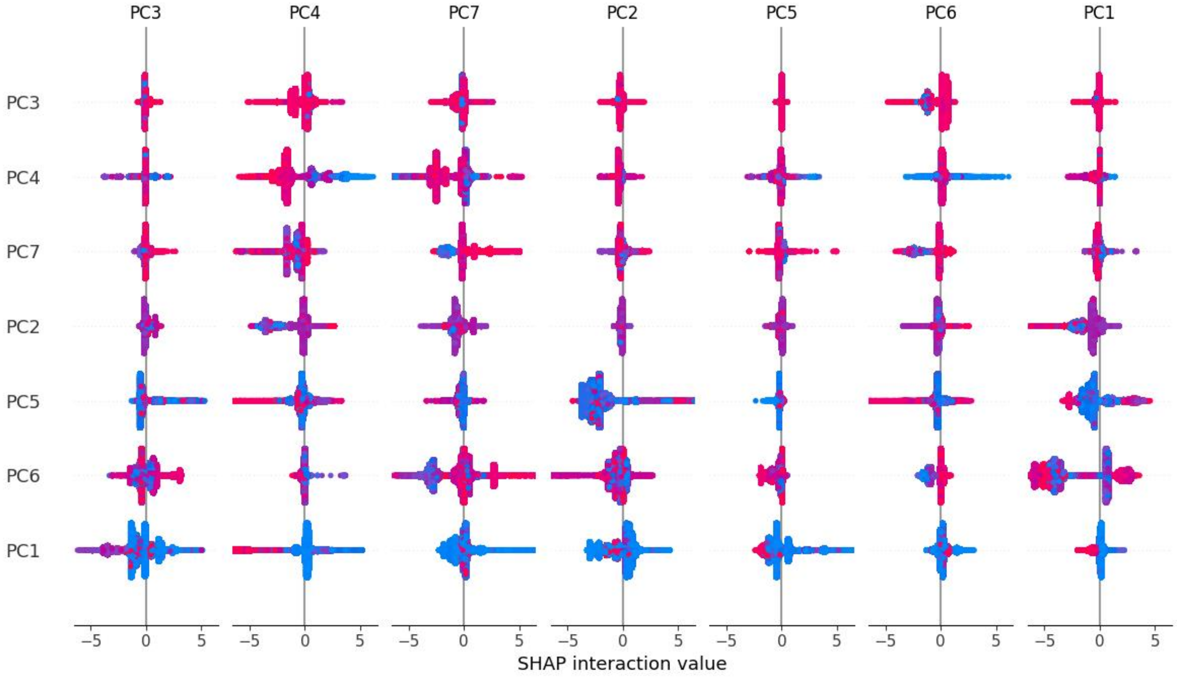


Figure 12: SHAP Interaction Summary Plot For XGB

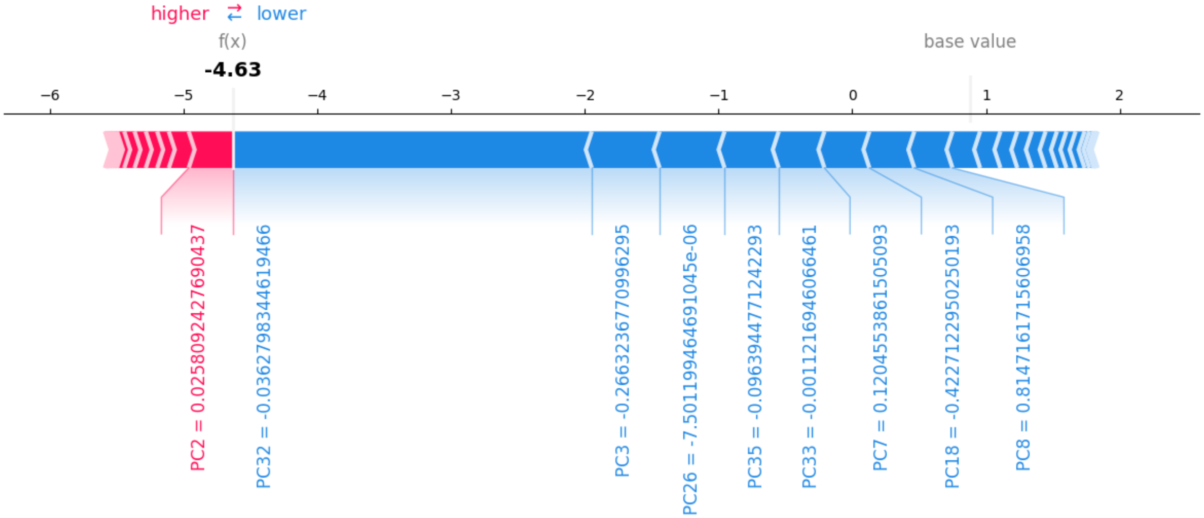


Figure 13: SHAP Waterfall Plot For XGB

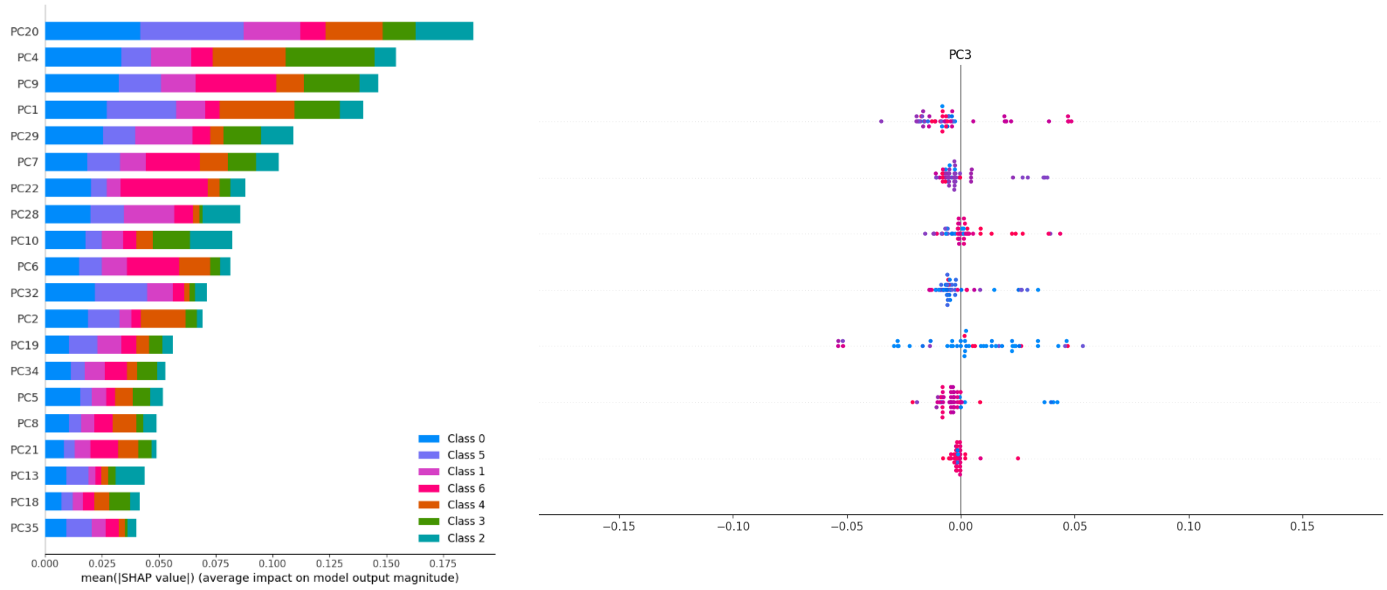


Figure 14: SHAP Graphs For Random Forest

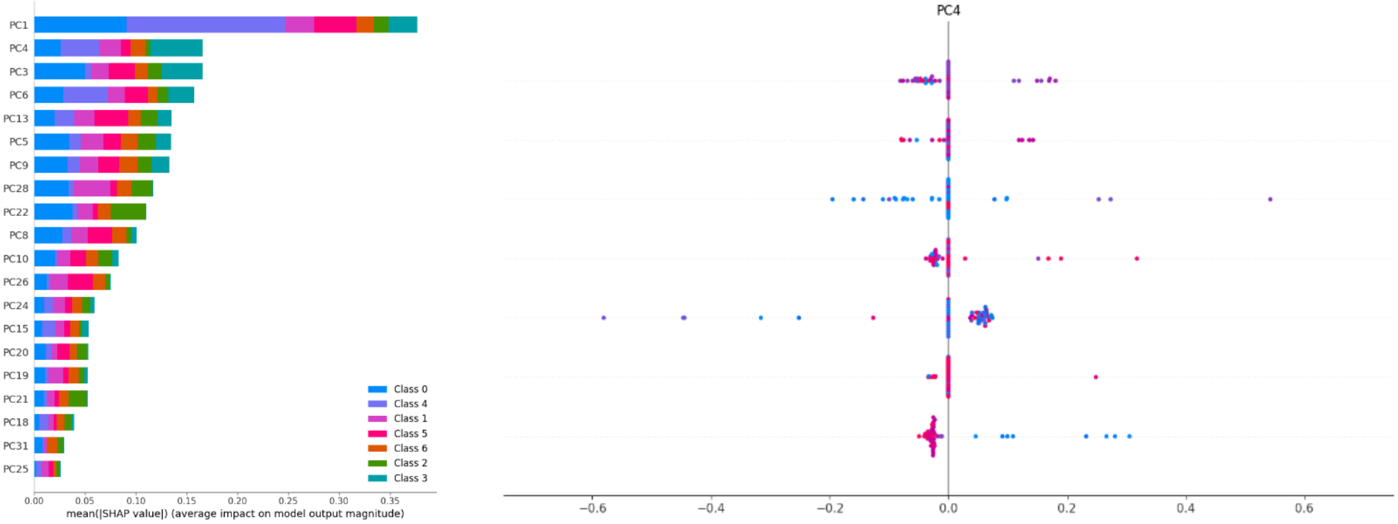


Figure 15: SHAP Plots For KNN

IV. Discussion

Our study aimed to evaluate different machine learning models for network intrusion and botnet detection while also emphasizing the importance of explainability in cybersecurity. The results provide valuable insights into how different approaches perform under varying conditions and highlight the strengths and weaknesses of each method.

5.1 Network Intrusion Detection (CICIDS Dataset)

The comparison of various models with the CICIDS dataset indicated that the Hybrid model (KNN with Random Forest) was the most effective, with a remarkable accuracy rate of 99.9%. This achievement is attributed to the complementary strengths of KNN's capability to capture local patterns and Random Forest's capability to deal with complex feature interactions. These findings align with previous studies that have explored hybrid approaches for encrypted traffic detection, emphasizing their effectiveness in identifying malicious patterns in network traffic [3], [7].

Additionally, the XGBoost models—most notably the one optimized with Bayesian optimization—were particularly powerful, demonstrating the benefits of boosting methods and hyperparameter tuning. Studies using deep learning for network anomaly detection, such as the implementation of LSTM Autoencoders for IoT botnet detection [11], further reinforce the need for feature extraction and optimization techniques to improve accuracy. In contrast, the standard Random Forest model, despite widespread acknowledgement of its power, exhibited marginal underperformance due to its fixed hyperparameters and low flexibility which is consistent with findings in feature-based decision tree approaches that suffer from generalization issues when not carefully tuned [1]. These results affirm the efficiency of ensemble learning and optimization methods in enhancing network intrusion detection systems.

5.2 Botnet Detection (CTU-NCC Dataset)

When detecting botnet activity, stacking-based ensemble models consistently outperformed voting-based approaches, with Decision Tree Stacking achieving the highest accuracy of 99.68%. This result highlights the power of stacking in refining predictions, especially for weaker learners like individual Decision Trees, which tend to overfit. Similar stacking techniques have been explored for botnet detection using GNN-based methods [12] and hybrid feature selection strategies [4], both demonstrating improvements in classification accuracy.

Interestingly, the Random Forest models performed well both with stacking and voting, a testament to their consistency in application to cybersecurity. The KNN models, even with decent accuracy rates (~97%), performed relatively poorly compared to tree-based methods, most likely due to them being data distribution-sensitive and computationally expensive. This is consistent with prior research indicating that KNN struggles with computational efficiency and data distribution sensitivity in network-based anomaly detection tasks [13]. Moreover, studies applying ResNet-18 for botnet classification [14] suggest that deep learning models may also struggle with encrypted traffic, requiring adaptations for improved performance. The findings suggest that while stacking is generally beneficial, its impact is most pronounced on models like Decision Trees, which benefit significantly from an ensemble approach.

5.3 The Importance of Explainability in AI

Apart from achieving high accuracy, our focus went beyond that to making the model decisions more interpretable using Explainable AI (XAI) techniques, namely SHAP and LIME. These techniques allowed us to determine the key factors influencing model predictions, such as flow duration, packet length variance, and port usage. By identifying feature importance, we were able to improve trust in the model’s predictions and reduce false positives—an approach supported by previous studies that emphasize the role of explainability in cybersecurity [1], [15]. By exposing the contribution of different features to classification labels, we were able to gain a better understanding of why certain network activities were being classified as attacks. This not only added to the model trust but also allowed us to further improve the model to reduce false positives and make better decisions.

V. Conclusion and Future Scope

6.1 Conclusion

In this current study, we compared a variety of machine learning algorithms for identifying network intrusions and botnets and compared their accuracy based on the CICIDS and CTU-NCC datasets. The results showed that ensemble learning algorithms, such as stacking and hybrid models, significantly improved the accuracy of classification. The Hybrid KNN-Random Forest model attained the best accuracy (99.9%) for intrusion detection, while the Decision Tree Stacking model performed better than all the other models (99.68%) for botnet detection. The results reflect the importance of combining heterogeneous algorithms to provide effective representation for different data patterns and improve the robustness of the model.

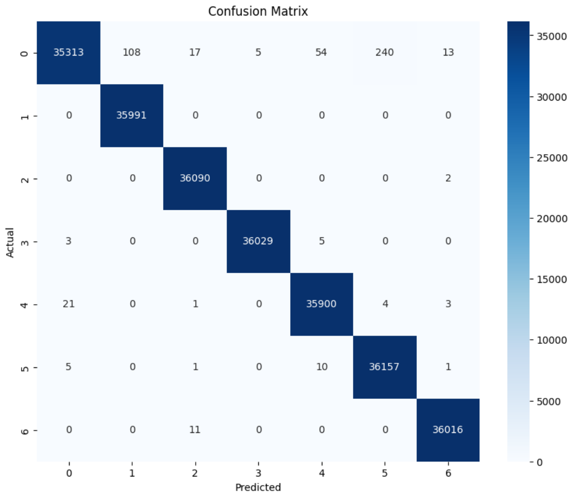


Figure 16: Confusion Matrix Of Hybrid Model

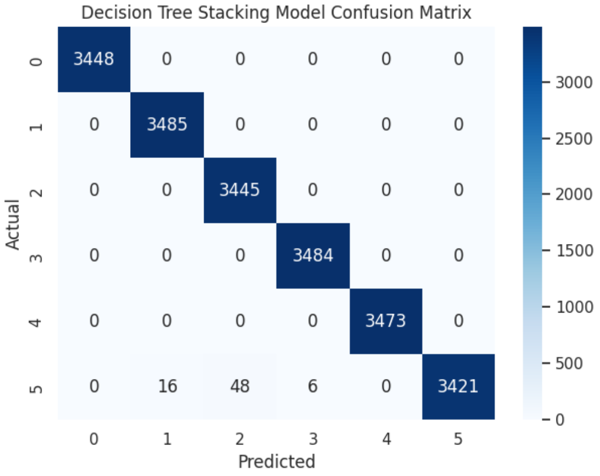


Figure 17: Confusion Matrix Of Decision Tree Stacking

In addition, we used Explainable AI (XAI) techniques, such as SHAP and LIME, to offer explanations of the reasons behind model outcomes. This helped us identify relevant features that impact predictions, hence enhancing the transparency and credibility of the model. By bridging the accuracy vs. transparency gap, we rendered our models not only effective but also understandable—a vital factor in the field of cybersecurity applications.

In short, our study emphasizes the significance of ensemble learning, hyperparameter optimization, and model interpretability as fundamental ingredients in building secure AI-driven security solutions. The fusion of robust classification techniques and explanation techniques is a solid foundation for effective cybersecurity applications, thus ensuring high performance as well as stability in identifying network threats.

6.2 Future Scope

While our study achieved promising results, several areas can be explored further to **enhance model performance and applicability in real-world scenarios**:

1. **Real-Time Threat Detection:**
   - Implementing the models in **real-time** network monitoring systems to detect intrusions as they occur.
   - Optimizing model inference speed for **low-latency** threat detection.
2. **Adaptive and Online Learning Models:**
   - Developing models that can continuously **learn and adapt** to evolving cyber threats.
   - Incorporating **reinforcement learning** or **online learning techniques** to dynamically update model parameters.
3. **Integration with Deep Learning:**
   - Combining **deep learning architectures** such as **transformers or LSTMs** with traditional machine learning to capture complex attack patterns.
   - Exploring **CNNs for packet-level analysis** in intrusion detection.
4. **Deployment in Large-Scale Networks:**
   - Testing the models on **enterprise-level** and **cloud-based** network infrastructures.
   - Evaluating performance on **heterogeneous datasets** with diverse network traffic.

By advancing in these directions, **AI-driven network security solutions can become more robust, scalable, and capable of adapting to new attack vectors**, ensuring **continuous protection against cyber threats in real-world environments**.

VI. References

[1] D. Zhao *et al.*, “Botnet detection based on traffic behavior analysis and flow intervals,” *Comput Secur*, vol. 39, no. PARTA, pp. 2–16, 2013, doi: 10.1016/j.cose.2013.04.007.

[2] C. Wei, G. Xie, and Z. Diao, “A lightweight deep learning framework for botnet detecting at the IoT edge,” Jun. 01, 2023, *Elsevier Ltd*. doi: 10.1016/j.cose.2023.103195.

[3] Z. Wang, K.-W. Fok, and V. L. L. Thing, “Machine Learning for Encrypted Malicious Traffic Detection: Approaches, Datasets and Comparative Study,” Mar. 2022, doi: 10.1016/j.cose.2021.102542.

[4] M. A. Hossain and M. S. Islam, “A novel hybrid feature selection and ensemble-based machine learning approach for botnet detection,” *Sci Rep*, vol. 13, no. 1, Dec. 2023, doi: 10.1038/s41598-023-48230-1.

[5] I. Sharafaldin, A. Habibi Lashkari, and A. A. Ghorbani, “Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization,” in *Proceedings of the 4th International Conference on Information Systems Security and Privacy*, SCITEPRESS - Science and Technology Publications, 2018, pp. 108–116. doi: 10.5220/0006639801080116.

[6] M. A. R. Putra, D. P. Hostiadi, and T. Ahmad, “Botnet dataset with simultaneous attack activity,” *Data Brief*, vol. 45, p. 108628, Dec. 2022, doi: 10.1016/j.dib.2022.108628.

[7] Z. Wang and V. L. L. Thing, “Feature Mining for Encrypted Malicious Traffic Detection with Deep Learning and Other Machine Learning Algorithms,” Apr. 2023, doi: 10.1016/j.cose.2023.103143.

[8] X. Meng, B. Lang, Y. Liu, and Y. Yan, “Deeply fused flow and topology features for botnet detection based on a pretrained GCN.”

[9] A. K. Kumar *et al.*, “Enhanced Hybrid Deep Learning Approach for Botnet Attacks Detection in IoT Environment,” in *2024 7th International Conference on Signal Processing and Information Security (ICSPIS)*, IEEE, Nov. 2024, pp. 1–6. doi: 10.1109/ICSPIS63676.2024.10812621.

[10] N. Moustafa and J. Slay, “UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set),” in *2015 Military Communications and Information Systems Conference (MilCIS)*, IEEE, Nov. 2015, pp. 1–6. doi: 10.1109/MilCIS.2015.7348942.

[11] S. I. Popoola, B. Adebisi, M. Hammoudeh, G. Gui, and H. Gacanin, “Hybrid Deep Learning for Botnet Attack Detection in the Internet-of-Things Networks,” *IEEE Internet Things J*, vol. 8, no. 6, pp. 4944–4956, Mar. 2021, doi: 10.1109/JIOT.2020.3034156.

[12] F. Alizadeh and M. Khansari, “An Analysis of Botnet Detection Using Graph Neural Network,” in *2023 13th International Conference on Computer and Knowledge Engineering (ICCKE)*, IEEE, Nov. 2023, pp. 491–495. doi: 10.1109/ICCKE60553.2023.10326235.

[13] X. Zang, T. Wang, X. Zhang, J. Gong, P. Gao, and G. Zhang, “Encrypted malicious traffic detection based on natural language processing and deep learning,” *Computer Networks*, vol. 250, Aug. 2024, doi: 10.1016/j.comnet.2024.110598.

[14] F. Hussain *et al.*, “A Two-Fold Machine Learning Approach to Prevent and Detect IoT Botnet Attacks,” *IEEE Access*, vol. 9, pp. 163412–163430, 2021, doi: 10.1109/ACCESS.2021.3131014.

[15] A. A. korba, A. Diaf, and Y. Ghamri-Doudane, “AI-Driven Fast and Early Detection of IoT Botnet Threats: A Comprehensive Network Traffic Analysis Approach,” Jul. 2024, [Online]. Available: http://arxiv.org/abs/2407.15688