EXPLAINABLE AI FOR INTRUSION DETECTION SYSTEM

THESIS SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

M.Sc COMPUTER SCIENCE

LIBA MARIYAM K

Reg.No: 2200705013

UNDER THE GUIDANCE OF

Dr. MANOHAR NAIK S

DEPARTMENT OF COMPUTER SCIENCE
SCHOOL OF PHYSICAL SCIENCES
CENTRAL UNIVERSITY OF KERALA
TEJASWINI HILLS, PERIYE, KASARAGOD - 671320,
KERALA, INDIA,
JUNE 2024

Department of Computer Science, Central University of Kerala
Tejaswini Hills, Periye - 671320, Kasaragod

CERTIFICATE

This is to certify that the thesis entitled, **"Explainable AI for Intrusion Detection System"** submitted by **Liba Mariyam K (REG. NO: 2200705013)** in partial fulfillment of the requirements for the award of M.Sc in Computer Science at the Central University of Kerala, is an authentic work carried out by her under my supervision and guidance.

To the best of my knowledge, the matter embodied in the report has not been submitted to any other University Institute for the award of any degree.

DATE : 13.06.2024.

DR. MANOHAR NAIK S
Assistant Professor
Department of Computer Science
Central University of Kerala
Kasaragod, Kerala - 671320

डॉ. मनोहर नायक. एस / Dr. Manohar Naik S
सहायक आचार्य / Assistant Professor
कंप्यूटर विज्ञान विभाग / Department of Computer Science
भौतिक विज्ञान स्कूल / School of Physical Sciences
केरल केंद्रीय विश्वविद्यालय / Central University of Kerala
तेजस्विनी हिल्स, पेरिय / Tejaswini Hills, Periya P.O
कासरगोड / Kasaragod-671316

Department of Computer Science, Central University of Kerala
Tejaswini Hills, Periye - 671320, Kasaragod

CERTIFICATE

This is to certify that the thesis entitled, **"Explainable AI for Intrusion Detection System"** is a bonafide work carried out by **Liba Mariyam K (REG. NO: 200705013)** in partial fulfillment of the requirements for the award of M.Sc in Computer Science at the Central University of Kerala, Kasaragod, during the academic year **2023-2024**.

The work is satisfactory to award Master's Degree in Computer Science.

DATE :

EXAMINER 1:

EXAMINER 2:

डॉ. आदित्या वी / Dr. Adithya V.
सहायक आचार्य / Assistant Professor
कम्प्यूटर विज्ञान विभाग / Department of Computer Science
भौतिक विज्ञान स्कूल / School of Physical Sciences
केरल केन्द्रीय विश्वविद्यालय / Central University of Kerala
तेजस्विनी हिल्स, पेरिया / Tejaswini Hills, Periya
कासरगोड / Kasaragod

13/6/2024

Dr. JAYASUDHA J S
Professor
Head Of the Department
Department of Computer Science
Central University of Kerala

DECLARATION

I, Liba Mariyam K , Reg No: 2200705013, student of Fourth Semester M.Sc Computer Science, Central University of Kerala, do hereby declare that the report entitled, **"Explainable AI for Intrusion Detection System"**, submitted to the Department of Computer Science is an original record of studies and bonafide work carried out by me from JAN 2024 to JUN 2024.

DATE: 13/06/2024

PLACE: Periye

LIBA MARIYAM K

2200705013

ACKNOWLEDGEMENT

Words play a heralding role in expressing our gratitude, if considered as a token of approval and symbol of acknowledgment. In this project, I have taken efforts. However, without the kind support and help of many individuals, it would not have been possible. I would like to extend my sincere thanks to all of them.

First and foremost, I express my profound gratitude to my guide, Dr. Manohar Naik S, Assistant Professor in the Department of Computer Science at Central University of Kerala.I am indebted to him for his invaluable support, constant encouragement, and constructive criticism that have been instrumental in shaping the course of this project. His guidance has been a driving force that propelled me forward, and I am deeply grateful for the motivation and direction he provided. Without his insightful mentorship, this project would not have come to fruition. I also convey my heartful thanks to the Head of the Department, Dr. Jayasudha J S, for providing access to the laboratory and research facilities. Without her precious support it would not be possible to conduct this research.

I also thank our Teachers, Scholars, and Friends of the Department of Computer Science for sharing their knowledge and suggestions with me. I would like to thank various websites, research papers, and resources from the internet that have contributed to my understanding and informed my work. While too numerous to list individually, I deeply appreciate their collective impact.

I am truly grateful for the divine grace that guided and empowered me to complete this work. Without God's blessings, this achievement would not have been possible.

LIBA MARIYAM K

2200705013

Abstract

In the realm of cybersecurity, intrusion detection systems (IDS) play a crucial role in preventing and mitigating threats, safeguarding computer networks by employing diverse machine learning methods. This paper investigates the development of an intrusion detection system (IDS) that leverages machine learning ensemble methods and Explainable Artificial Intelligence (XAI) techniques. By employing decision trees, random forests, and support vector machines, the proposed IDS aims to achieve high classification accuracy (98.47%) while minimizing false positives on the CICIDS-2017 dataset. The integration of LIME (Local Interpretable Model-agnostic Explanations) within the XAI framework fosters interpretability and transparency, addressing the "black-box" nature of traditional IDS models. This project explores the effectiveness of LIME in enhancing model understanding and decision-making processes within the IDS. The findings contribute to the development of more robust, transparent, and trustworthy AI-powered security solutions.

Table of contents

List of figures

List of tables

Chapter 1

INTRODUCTION

In the digital age, the world is increasingly interconnected, with vast amounts of data being generated and shared across various platforms. This connectivity brings immense opportunities but also poses significant challenges, particularly in the realm of cybersecurity. Cyber threats are evolving rapidly, targeting everything from personal devices to critical infrastructure, posing substantial risks to privacy, security, and economic stability. Intrusion Detection Systems (IDS) play a pivotal role in mitigating these threats by identifying suspicious activities and preventing unauthorized access to sensitive information [1].

Traditional IDS often rely on rule-based systems or anomaly detection methods, which, while effective, struggle with the complexity and diversity of modern cyber threats. The advent of machine learning (ML) has introduced powerful tools capable of learning from past incidents to predict and prevent future attacks. However, the effectiveness of these ML-based IDS is limited by their lack of transparency and explainability, making it challenging for stakeholders to understand how decisions are made and why certain actions are recommended.

This project delves into the intersection of machine learning and cybersecurity, proposing an innovative IDS leveraging ensemble learning techniques and Explainable AI (XAI) to enhance both the accuracy and transparency of intrusion detection. By employing a combination of decision trees, random forests, and Support Vector Machines (SVM), alongside the XAI algorithm LIME, we aim to create a system that not only excels in identifying threats but also offers clear insights into its decision-making process. Our focus on the CICIDS-2017 dataset [2], a comprehensive resource for evaluating IDS, underscores our commitment to developing a solution that addresses real-world cybersecurity challenges.

Through this project, we aspire to contribute to the ongoing efforts to fortify digital defenses, ensuring a safer online environment for individuals and organizations alike. By

bridging the gap between the technical intricacies of machine learning and the practical needs of cybersecurity, we hope to pave the way for more transparent and effective intrusion detection systems.

1.1 Background to the project

As cyberattacks become more sophisticated, traditional methods of protecting our networks are struggling to keep up. Intrusion Detection Systems (IDS) are like security guards, constantly scanning our digital doors for suspicious activity. However, traditional IDS methods, reliant on pre-defined rules, often struggle to adapt to the ever-evolving tactics of cyber attackers. This limitation can lead to a high number of false positives, wasting valuable time and resources, or worse, missed threats that leave our systems vulnerable.

This project tackles this challenge by harnessing the power of Machine Learning (ML). ML algorithms can learn from vast amounts of data, enabling them to identify patterns and anomalies that might signify an intrusion attempt. This allows for a more dynamic and adaptable approach to intrusion detection, significantly reducing false positives and improving overall network resilience. However, complex ML models often operate as "black boxes," making it difficult to understand how they reach specific decisions. This lack of transparency can hinder trust and limit the real-world implementation of these powerful tools.

1.2 Motivation to the project

The motivation for this project stems from two key concerns:

- The Evolving Threat Landscape: Cybercriminals are constantly developing new and sophisticated attack methods. Traditional IDS, reliant on static rules, struggle to keep pace with this rapid evolution. This leaves our networks vulnerable to undetected breaches and potential data loss.
- The Black Box Problem: While ML offers a powerful solution for intrusion detection, its complex models can be opaque. We lack insight into how these models arrive at their decisions, hindering trust and hindering the practical application of these valuable tools [3].

This project aims to address both these issues. By combining ML with Explainable Artificial Intelligence (XAI), we can create an IDS that is not only effective in detecting

modern threats but also transparent in its decision-making process. This transparency fosters trust and allows security personnel to understand the reasoning behind the system's alerts, enabling them to respond more effectively to potential security breaches.

1.3 Organization of Thesis

The thesis is organized as follows. Chapter 2 is about the literature reviews on the area of Intrusion Detection System and Explainable Artificial Intelligence. Chapter 3 discusses the methodology of the proposed paper. Description of dataset , preprocessing of datasets, feature extraction & classification are the various stages of methodology. Chapter 4 is about the experimental analysis and results. Chapter 5 lists the papers referred to for doing this work.

Chapter 2

LITERATURE REVIEW

2.1 Introduction

In this study, this chapter addresses prior research efforts aimed at enhancing Intrusion Detection Systems (IDS) through machine learning (ML) and deep learning (DL) techniques. It encompasses studies exploring novel intrusion detection methods, addressing challenges in ML safety, and proposing frameworks for fairness and robustness in federated learning systems. The literature review is organized into two sections: the first focuses on advancements in IDS using ML and DL techniques, examining the effectiveness of various algorithms and models; the second section emphasizes explainability and transparency in AI models for IDS, highlighting the importance of interpretability and trust in cybersecurity applications. These studies collectively contribute to advancing IDS by addressing key issues such as accuracy, robustness, and real-world applicability, laying the groundwork for future research in the field.

2.2 Intrusion Detection with ML DL Techniques

Peddabachigari et al. explored new intrusion detection techniques using hybrid intelligent systems and evaluated their performance on the KDD Cup 99 dataset [4]. Their

study focused on data temporal correlation (DT) and sparse maximum likelihood (SVM) methods, as well as a hybrid DT-SVM model and an ensemble approach. Results showed that DT performed equally well or better than Probe, U2R, and R2L for all classes, while the hybrid DT-SVM approach outperformed direct SVM. The ensemble approach achieved 100% accuracy for the Probe class, indicating potential for other classes to reach similar levels with appropriate base classifiers. They proposed a hierarchical intelligent IDS model to optimize intrusion detection using the best-performing individual classifiers and ensemble methods.

A study on intrusion detection systems focused on evaluating shallow and deep neural networks for cybersecurity. The research employed deep neural networks with a learning rate of 0.1 and 1000 iterations, using the KDD CUP 99 dataset to enhance intrusion detection accuracy. By comparing various classical machine learning techniques, it was found that deep neural networks with three layers, after 100 iterations, outperformed other methods in terms of accuracy. Additionally, the research proposed an explainable AI framework incorporating transparency capabilities throughout the machine learning pipeline, employing techniques such as SHAP, LIME, CEM, ProtoDash, and Boolean Decision Rules. These approaches were applied to the NSL-KDD dataset, enhancing the explainability of intrusion detection systems [5].

Hoque et al. developed an Intrusion Detection System (IDS) to find network intrusions. They used a technique called genetic algorithms (GA) to make this system effective. With the growing need to secure data on networks, IDS has become crucial for computer and network safety. Despite various methods used in intrusion detection, none are perfect, so researchers keep trying to make them better. Hoque et al.'s IDS uses GA to efficiently spot intrusions by using evolution theory to sort through traffic data and make it simpler. They explained how they set up and used GA in detail. They tested their system using the KDD99 dataset, a standard dataset for this purpose, and found it could

detect intrusions reasonably well. They also looked into using different machine learning techniques, including deep neural networks, on the same dataset, showing a thorough approach to intrusion detection [6].

Another comprehensive review of anomaly-based intrusion detection systems (AIDS), addressing issues found in related literature, such as randomness in algorithm selection and parameterization, outdated datasets, and shallow result analyses [7] . They evaluated 10 popular supervised and unsupervised machine learning algorithms, including artificial neural networks, decision trees, k-nearest neighbors, and support vector machines, among others, using the CICIDS2017 dataset. Unlike previous studies, they assessed AIDS performance using metrics like true positive and negative rates, accuracy, precision, recall, and F-Score for 31 ML-AIDS models. The study also considered training and testing times, recognizing the importance of time complexity in AIDS. Results showed that k-NN, decision tree, and naive Bayes models performed exceptionally well, particularly in detecting web attacks, showcasing their effectiveness in real-world network security scenarios.

A technical review and comparative analysis of machine learning techniques for intrusion detection systems in manet comparing three deep learning models - Inception-CNN, BLSTM, and DBN - for Intrusion Detection Systems (IDS) in mobile ad hoc networks (MANETs) using the NSL-KDD dataset [8]. Their research focused on the methods and findings of these models in detecting and classifying cyber attacks. They found that Inception-CNN outperformed the other models, demonstrating superior ability to learn normal behavior from large datasets and effectively detect unseen threats. This study underscores the potential of deep learning techniques in enhancing the accuracy and reliability of IDS in dynamic network environments like MANETs.

A deep learning-based approach was developed for a flexible and efficient Network Intrusion Detection System (NIDS) to tackle unforeseen and unpredictable attacks. Using Self-taught Learning (STL) on the NSL-KDD dataset, they implemented a sparse autoencoder

and soft-max regression-based NIDS, achieving strong anomaly detection accuracy compared to previous implementations. They suggested enhancements using techniques like Stacked Autoencoder for unsupervised feature learning and classifiers like NB-Tree, Random Tree, or J48. Future research envisions a real-time NIDS for actual networks using deep learning and exploring on-the-go feature learning from raw network traffic headers [9] .

Yulianto et al. [10] explore methods to enhance the performance of AdaBoost-based Intrusion Detection Systems (IDS) on the CIC IDS 2017 Dataset. They employ Synthetic Minority Oversampling Technique (SMOTE), Principal Component Analysis (PCA), and Ensemble Feature Selection (EFS) to address data imbalance and select important attributes. Their approach, validated through evaluation, yields promising results with an accuracy of 81.83%, precision of 81.83%, recall of 100%, and F1 Score of 90.01%. This research contributes to improving IDS performance in challenging datasets, highlighting its potential for real-time application in data security frameworks.

A proposed network intrusion detection algorithm addresses the challenge of data imbalance in training models by combining an enhanced random forest with the synthetic minority oversampling technique (SMOTE) to balance the dataset and improve learning of minor sample features. The hybrid approach incorporates K-means clustering and SMOTE sampling to increase the number of minor samples, and prediction results are corrected using a similarity matrix of network attacks to enhance detection accuracy. Testing on the NSL-KDD dataset yielded a classification accuracy of 99.72% on the training set and 78.47% on the test set, showing promising improvements in detection accuracy. This study highlights the potential of enhancing random forest algorithms for intrusion detection systems and suggests future research should focus on optimizing model accuracy and computational efficiency through feature extraction and classifier selection [11].

Mulay, et al. introduce a novel approach for multiclass intrusion detection using a decision-tree-based support vector machine (SVM) algorithm [12] . The algorithm combines

SVM and decision tree models to efficiently classify data, reducing training and testing time while improving system efficiency. By constructing binary trees that divide the dataset into subsets based on class, the algorithm optimizes classification performance. Although final results are pending, preliminary findings suggest that the integrated model outperforms individual models, offering promising potential for faster and more effective intrusion detection systems.

2.3 Explainability and Transparency in AI Models for IDS

Additionally researchers discussed challenges and opportunities in ensuring machine learning (ML) safety for open-world tasks. They compared ML algorithms' limitations in uncontrolled open-world scenarios with conventional safety standards, emphasizing the need for runtime error detection. These strategies aim to enhance ML dependability, improve model performance and robustness, and detect errors during runtime. ML safety research focuses on mitigating long-term risks associated with ML, particularly in cases where ML capabilities surpass safety measures or when safety challenges are anticipated to increase in complexity in the next decade [13].

The Ditto framework for achieving fairness and robustness in federated learning systems analysed by Li et al. [14]. They found that heterogeneous networks pose challenges where robustness to attacks and fairness compete for resources. The Ditto framework, along with a scalable solver, addresses these challenges by analyzing linear problems to determine if fairness and robustness can be achieved simultaneously. Their empirical findings show that Ditto performs competitively compared to recent methods and produces more accurate, robust, and fair models compared to standard baselines.

A comprehensive review of network intrusion detection mechanisms, focusing on machine learning (ML) and deep learning (DL) methods, elucidated intrusion detection

systems (IDS) and provided a taxonomy based on notable ML and DL techniques used in network-based IDS (NIDS) systems [15]. They discussed the strengths and limitations of various solutions, revealing a trend towards DL-based methodologies to enhance NIDS performance, with around 80% using DL approaches like autoencoder (AE) and deep neural networks (DNN). Despite DL schemes offering superior performance, they entail complexity and high computational resources. The study emphasized the need for modern datasets like CSE-CIC-IDS2018 for addressing contemporary network attacks and highlighted research gaps in improving model performance for low-frequency attacks and reducing complexity in proposed solutions.

Mukherjee et al. enhanced IDS efficiency by focusing on feature selection, investigating Correlation-based Feature Selection (CFS), Information Gain (IG), and Gain Ratio (GR), and proposing the Feature Vitality Based Reduction Method (FVBRM) [16]. Results showed CFS improved the Naïve Bayes classifier's accuracy over IG and GR, with IG outperforming GR despite being its extension. The FVBRM method further improved classification accuracy compared to CFS, though with increased computational time.

A proposed taxonomy categorized IDS based on data sources such as logs, packets, flow, and sessions, analyzing their suitability for detecting various attack types [17]. The study emphasized ML and DL techniques, particularly deep neural networks, to enhance IDS performance. Despite their superior fitting and generalization abilities, DL models' long running times may hinder real-time IDS operation. Challenges like limited datasets and the need for interpretable models were highlighted, offering insights into future IDS development trends.

An analysis of 37 cases explored the role of Intrusion Detection and Prevention Systems (IDPS) in defending against cyberattacks targeting Smart Grid (SG) infrastructure [18]. The study identified limitations in existing IDPS and recommended future research directions, emphasizing the need for IDPS to protect SG communications. It proposed

developing a Security Information and Event Management (SIEM) system tailored for SG, integrating Software-Defined Networking (SDN) technology and big data analytics to address identified deficiencies.

Wang et al. proposed a framework using SHapley Additive exPlanations (SHAP) to provide local and global explanations for IDS decisions [19]. Tested on the NSL-KDD dataset, the framework demonstrated its effectiveness in interpreting IDS decisions, identifying important features and their relationships with different attacks. The study compared interpretations between one-vs-all and multiclass classifiers, offering insights for designing more effective IDS structures, though further work is needed to utilize diverse datasets and achieve real-time interpretation.

A method to enhance the interpretability of "black box" models for bankruptcy prediction by infusing domain knowledge was proposed [20]. This approach improves the interpretability of models without significant performance compromise. By incorporating popular concepts like the 5 C's of credit, the method demonstrated improved explainability. Future work includes validating the approach with various datasets and enhancing interpretability by segregating each feature's contribution. The findings suggest potential applications in cybersecurity for better understanding and defending against attacks.

A comprehensive survey on Explainable Artificial Intelligence (XAI) methods for cybersecurity addressed the lack of transparency in conventional AI models used for cyber defense [21]. The review covered XAI models and their applications in defending against cyberattacks, providing insights into XAI-based defensive mechanisms across different sectors. Challenges, insights, and future research directions in XAI for cybersecurity were discussed, serving as a reference for researchers and professionals.

A review of XAI applications in cybersecurity highlighted the need for interpreting AI model decisions [22]. It discussed the necessity of XAI in understanding AI-generated results, particularly for predicting and analyzing attacks. The review covered traditional risk

management and the shift towards ML algorithms, discussing challenges like susceptibility to manipulation and resource-intensive development. XAI offers transparency and explainability, facilitating effective decision-making in cybersecurity. The study suggested fusion models using diverse data sources to optimize XAI benefits.

Kuppa et al. emphasized the importance of explanation methods in enhancing trust and understanding of black-box classifiers in cybersecurity [23]. They identified a gap in understanding how explanations can introduce new attack vectors and proposed a black-box attack leveraging XAI methods to compromise classifiers' confidentiality and privacy. The study demonstrated these attacks' effectiveness on various datasets and models, highlighting potential security threats from exposing explanations. Future research should explore different data types and model architectures and consider the tension between security and usability in cybersecurity explanations.

2.4 Summary

This chapter has reviewed significant advancements in IDS using machine learning and deep learning techniques, as well as the critical importance of explainability in AI models for IDS. The first section detailed various methodologies and their effectiveness in enhancing IDS performance, while the second section emphasized the need for transparency and interpretability in AI-driven cybersecurity solutions. Together, these studies underscore the ongoing efforts to improve IDS accuracy, robustness, and applicability in real-world scenarios, while also addressing the imperative of making AI decisions understandable and trustworthy.

Chapter 3

METHODOLOGY

This chapter outlines the methodology used to construct an intrusion detection system (IDS) that reliably discriminates between network data that is malicious and benign. The method consists of multiple stages: feature selection to determine the most important traffic characteristics; machine learning model training to provide reliable classification algorithms; and data preparation to address class imbalance. The system also incorporates LIME (Local Interpretable Model-Agnostic Explanations) for providing interpretative support for the IDS's judgments, hence augmenting interpretability and user trust. The diagrammatic representation of the proposed model is given in Fig. 3.1.

3.1 Dataset description

The CICIDS2017 dataset is a comprehensive collection of labeled network flows designed for intrusion detection research. It was created to reflect real-world network traffic, capturing both normal and malicious activities to provide a rich dataset for developing and testing intrusion detection systems.

One of the key features of the CICIDS2017 dataset is its variety of attack types, which includes web attacks, botnets, and insider threats, offering a broad spectrum of real-world

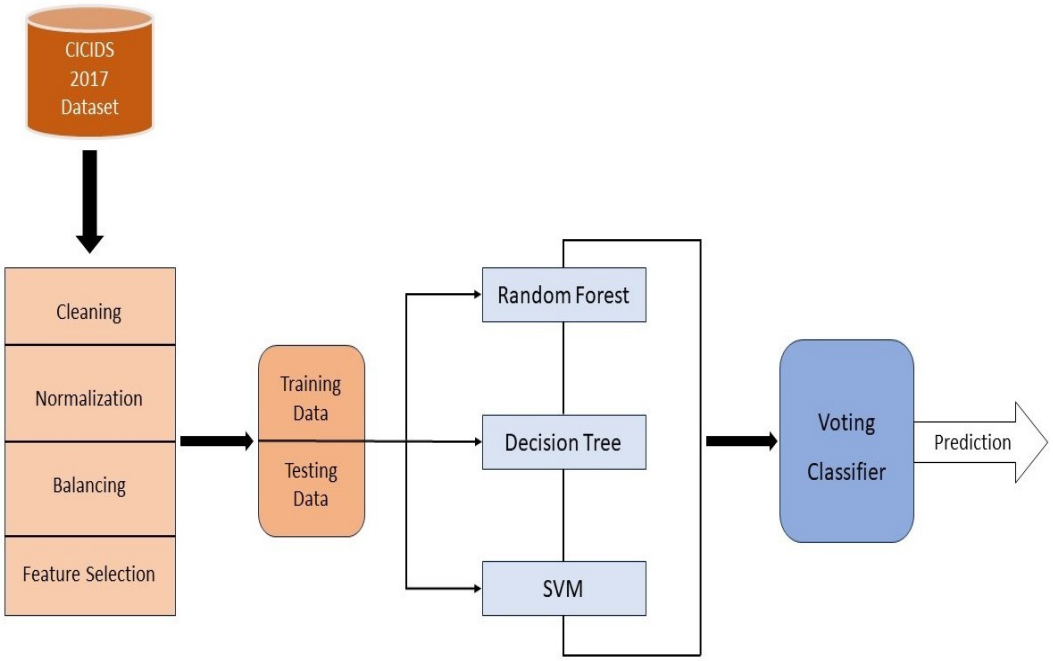

Fig. 3.1 Proposed System Flow Chart

scenarios. In addition to these attacks, the dataset captures regular network traffic to provide a baseline for detecting anomalies. The dataset is accompanied by comprehensive metadata, including timestamps, attack types, flow details, and labels, which facilitate informed analysis. Moreover, the dataset is publicly available for research purposes, encouraging collaboration and innovation in the field of intrusion detection.

Spanning five days (Figure 3.2), the dataset includes approximately 80 features and 15 unique classes, divided across eight CSV files, which can be concatenated to form a complete dataset. Despite its robustness, the CICIDS2017 dataset presents some challenges, such as the presence of NaN values that require preprocessing and an uneven distribution of classes that necessitates techniques to address potential biases. Additionally, the large volume of data requires substantial computational resources for processing and model training.

By leveraging the CICIDS2017 dataset, this project aims to develop a robust Intrusion Detection System (IDS) capable of effectively detecting a wide range of cyber

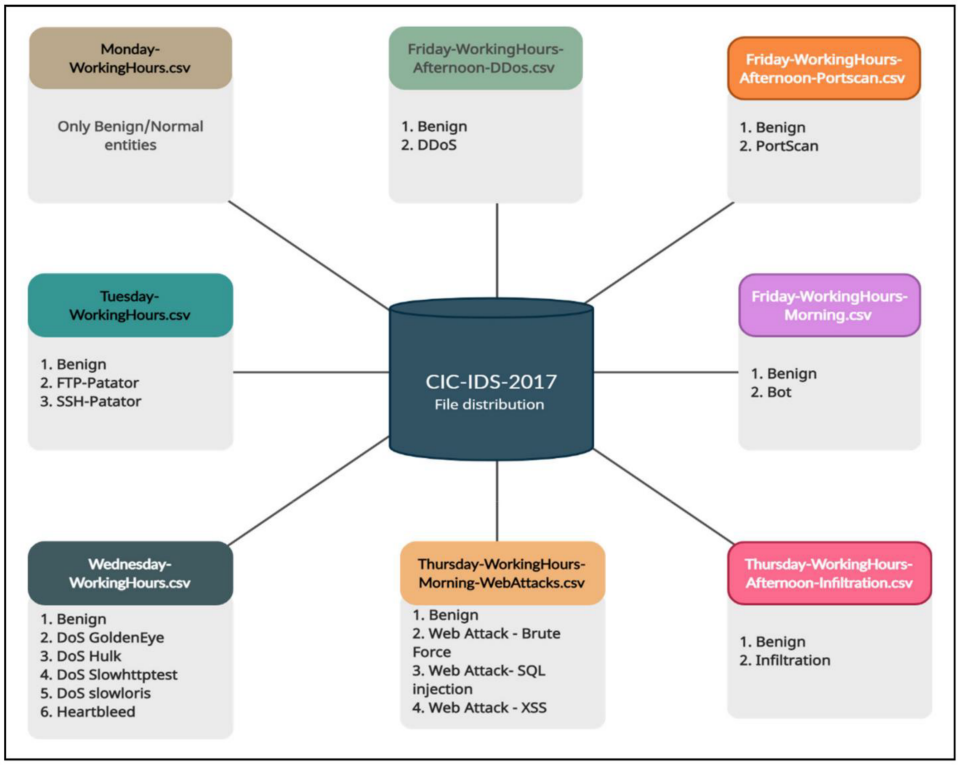

Fig. 3.2 File distribution of the dataset CIC-IDS-2017

threats. The dataset's comprehensive nature, realistic traffic patterns, and detailed labeling provide an invaluable foundation for training and evaluating IDS models.

3.2 Data Preprocessing

Before proceeding with the experiments, the CICIDS2017 dataset was processed and transformed into a structured format suitable for machine learning [24].The following data preprocessing steps were applied:

3.2.1 Cleaning Data

The initial stage of data preprocessing for the CICIDS2017 dataset involved meticulous cleaning to ensure data integrity and suitability for machine learning analysis. First, we ad-

dressed missing values by replacing infinite or very large values with ‘NaN‘ and subsequently dropping rows containing ‘NaN‘ values to maintain dataset consistency. We then removed columns with zero variance, as they offer no discriminatory power for classification.To address outliers, we utilized z-scores to identify data points that significantly deviated from the majority of the data, as visualized in the figure 3.3. These outliers were subsequently removed using a threshold-based approach, resulting in a cleaner dataset for subsequent analysis. These steps ensured a high-quality dataset, laying a solid foundation for subsequent normalization and model training phases.

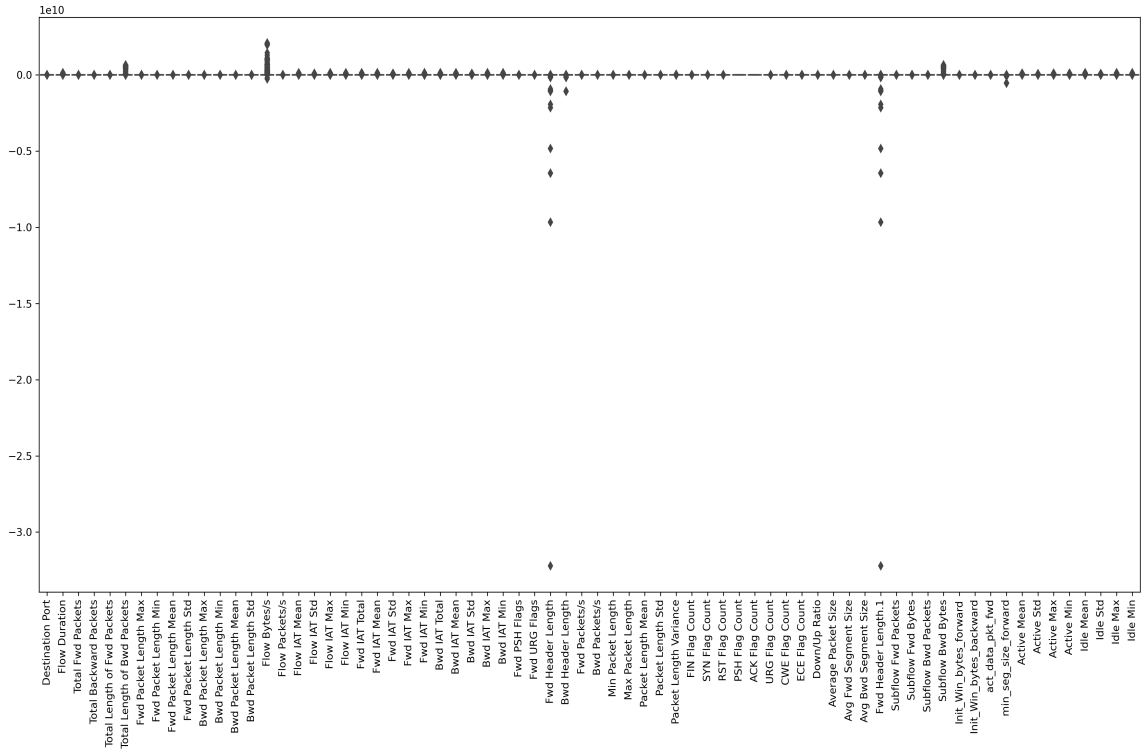

Fig. 3.3 Boxplot of outliers in Dataset

3.2.2 Normalization

Following data cleaning, normalization is crucial to ensure all features within the dataset are on a consistent scale, which prevents features with larger ranges from dominating

the learning process. In this project, we employed Min-Max Scaling to transform each feature value to a range between 0 and 1. This technique ensures that all features contribute equally to the model's learning process, regardless of their original unit or scale, thereby enhancing the effectiveness of the machine learning algorithms. Normalization improves the model's ability to learn from all features uniformly, leading to more accurate and reliable predictions.

3.2.3 Balancing Class Distribution

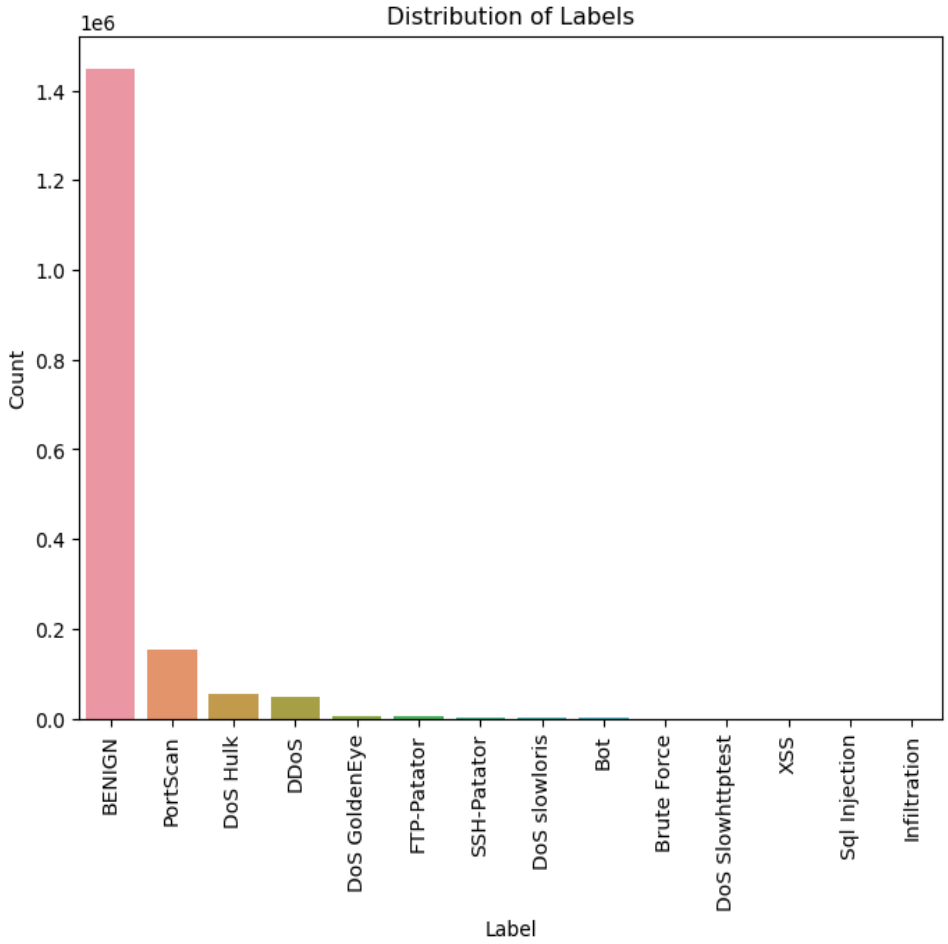

Fig. 3.4 Class distribution before balancing

The CICIDS2017 dataset, like many real-world datasets, presents class imbalance (figure 3.4), where certain attack types (minority classes) are outnumbered by benign network

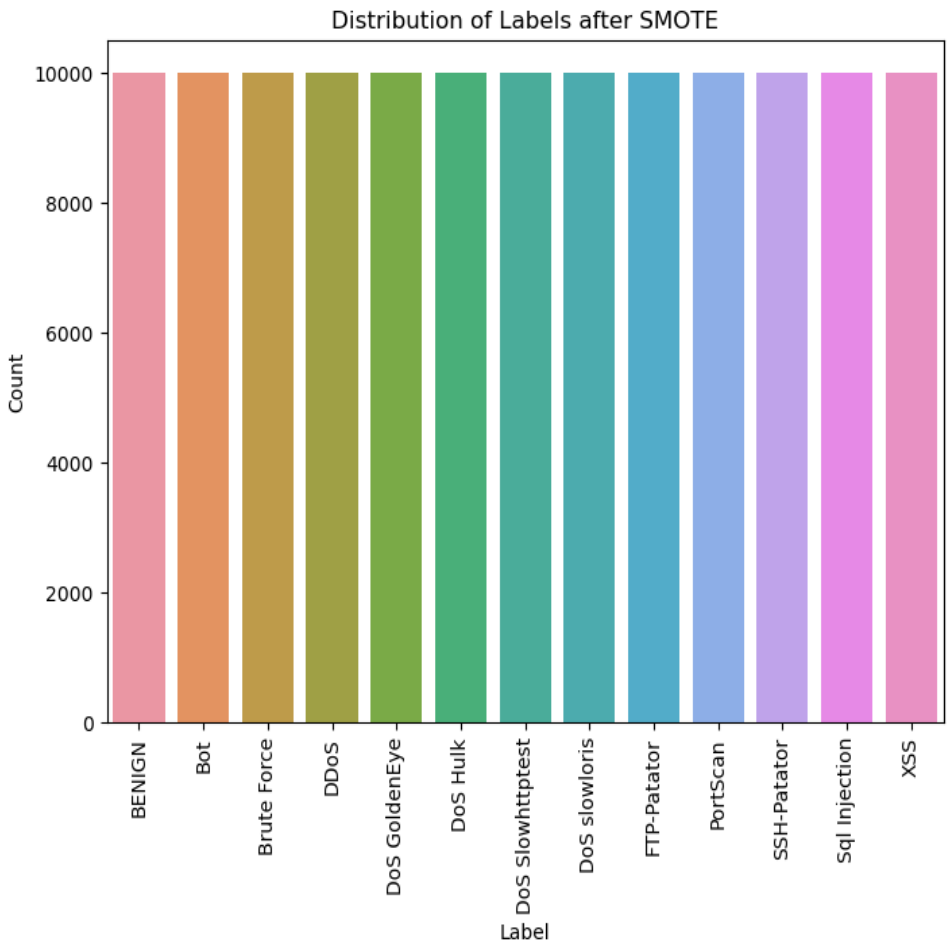

Fig. 3.5 Class distribution after balancing

traffic (majority class). Unaddressed class imbalance can bias machine learning models towards the majority class, potentially overlooking rare but crucial attack types.

To tackle this issue, our project adopts a dual strategy for data balancing. Firstly, we employ Random Under-Sampling using the ‘RandomUnderSampler‘ from the imbalanced-learn library [25]. This technique randomly removes data points from the majority class until each class has an equal number of instances, ensuring fair representation across all classes. Secondly, we integrate SMOTE (Synthetic Minority Over-Sampling Technique) to counteract the reduction in dataset size caused by under-sampling [26]. SMOTE synthesizes new data points for minority classes based on existing examples, effectively increasing their

representation within the dataset while preserving the original feature space.

By combining under-sampling and SMOTE, we create a balanced dataset (figure 3.5) where all attack types are proportionally represented. This balanced dataset facilitates a more accurate learning process, enabling the machine learning model to effectively discern between normal and malicious activity. The provided code snippet demonstrates the application of these techniques, including calculating the minimum number of samples across classes, defining oversampling ratios with SMOTE, and implementing the balancing process within a pipeline. Visualizations of the class distribution before and after balancing underscore the efficacy of these techniques in addressing class imbalance.

3.2.4 Feature Selection

we employed a feature selection technique based on a Random Forest classifier to identify the most informative features for intrusion detection. This approach leverages the inherent feature importance calculation within Random Forest models [27].

The process involved the following steps:

1. Random Forest Model Training: We trained a Random Forest classifier on the training data and target variable. This training process allows the model to learn the relationships between features and the attack types.

2. Feature Importance Extraction: Following model training, we retrieved the feature importances associated with each feature in the dataset. These importances indicate the relative contribution of each feature to the model's classification decisions.

3. Feature Ranking and Selection: Based on the extracted feature importances, we ranked the features in descending order of their importance. The top 10 features, were selected as the most relevant features for intrusion detection.

This feature selection approach leverages a trained Random Forest classifier to identify the most discriminative features for intrusion detection within the specific dataset. This data-driven method ensures the selected features are directly relevant to the task at hand, while the feature importances offer valuable insights into the model's decision-making process. These insights can be used to understand the model's behavior and potentially refine the feature set further. By focusing on the top 10 most important features (listed in Table 3.1), subsequent machine learning models can concentrate their learning on the most critical aspects of network traffic, leading to a potentially more robust and efficient intrusion detection system. The relative significance of these selected features is further illustrated in a feature importance chart 3.6, which visually depicts the contribution of each feature to the model's classification decisions.

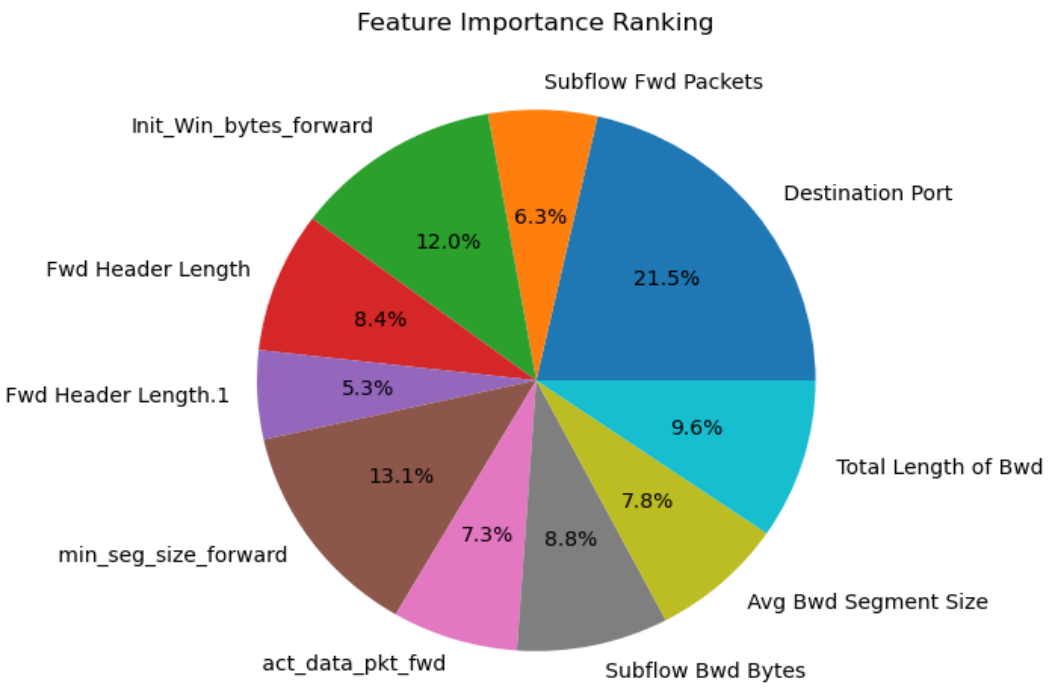

Fig. 3.6 Distribution of Feature Importance Scores

Table 3.1 Top 10 Features in Dataset

Rank	Features
1	Destination Port
2	Subflow Fwd Packets
3	min_seg_size_forward
4	Total Fwd Packets
5	Init_Win_bytes_forward
6	act_data_pkt_fwd
7	Fwd Header Length.1
8	Total Length of Bwd Packets
9	Fwd Header Length
10	Subflow Bwd Packets

By focusing on these key features, the machine learning model can concentrate its learning process on the most discriminative aspects of network traffic, leading to a more robust and efficient intrusion detection system. This meticulous feature selection ensures that the model's resources are allocated to the most informative features, enhancing its ability to accurately distinguish between benign and malicious network activity.

3.3 Model Training

Following the data preprocessing and balancing steps, this section delves into the training process and hyperparameter tuning for various machine learning models employed in the intrusion detection system using the CICIDS-2017 dataset. Each model offers unique advantages and considerations for effective intrusion detection.

3.3.1 Decision Tree (DT)

Decision Trees are tree-like models that iteratively split the data based on feature values, leading to leaf nodes representing specific classes (attack types). The DT model was trained on the prepared dataset, with a particular focus on hyperparameters such as the maximum tree depth and the minimum number of samples per leaf. By optimizing these parameters, we aimed to balance the model's complexity and prevent overfitting. This balance is crucial, as overly complex trees may memorize the training data rather than generalizing from it, while too simple trees might fail to capture necessary patterns. The interpretability

of decision trees (DTs) allows for easy visualization of decision-making processes, making them useful for understanding how specific features contribute to classification decisions and reflects how well the model performs on the test dataset [28].

3.3.2 Random Forest (RF)

Random Forests build on Decision Trees by creating an ensemble of trees, each trained on a random subset of features and data points. This ensemble approach reduces variance and improves robustness [29]. Our RF model was optimized using grid search to tune hyperparameters such as the number of trees (n_estimators) and the maximum depth of individual trees. This optimization ensured a diverse and robust collection of trees, enhancing the model's accuracy and reducing overfitting. Random Forests benefit from their ability to handle a large number of features and their robustness to noise. They also provide an internal mechanism for assessing feature importance, which can be valuable for understanding which features most significantly impact the model's predictions.

3.3.3 Support Vector Machine (SVM)

Support Vector Machines are powerful classifiers, particularly effective in high-dimensional spaces. We implemented an SVM model with a linear kernel suitable for our dataset [30]. Hyperparameter tuning was performed using grid search to optimize parameters such as the regularization parameter (C) and the kernel coefficient (gamma). These parameters were adjusted to maximize the margin between classes and minimize classification errors. SVMs work by finding the hyperplane that best separates different classes in the feature space, which is critical for accurate intrusion detection. Additionally, SVMs are known for their robustness in situations where the number of features exceeds the number of samples, making them particularly suited for high-dimensional network traffic data.

3.3.4 Voting Classifier

The Voting Classifier combines multiple models to enhance performance through ensemble learning. We integrated the DT, RF, and SVM models into a Voting Classifier, which aggregates predictions from each model using a majority voting scheme. This approach leverages the strengths of each individual model, aiming to achieve a more robust and accurate classification of network traffic. By combining models, the Voting Classifier can mitigate the weaknesses of individual models and capitalize on their strengths, leading to improved

intrusion detection performance. The flexibility of voting classifiers allows for either hard voting (majority class label) or soft voting (weighted average probabilities), providing further customization to optimize model performance.

The following section will present the evaluation results from training and testing these models. We will analyze their performance on the CICIDS-2017 dataset, providing insights into their effectiveness for intrusion detection and identifying potential areas for further improvement. Additionally, we will explore how the integration of these models into a comprehensive ensemble method like the Voting Classifier can lead to superior performance and reliability in real-world network security applications.

3.4 Generation of Explanations using LIME

Traditional machine learning models, particularly ensemble models like the one used in this IDS, can be complex and non-transparent in their decision-making processes [22]. This lack of interpretability can hinder trust in the IDS and make it difficult to understand why specific network traffic samples are classified as malicious (figure 3.7). XAI techniques address this challenge by providing insights into the model's reasoning behind its predictions (figure 3.8).

LIME (Local Interpretable Model-Agnostic Explanations) is a prominent XAI technique used to generate explanations for individual predictions made by any machine learning model, regardless of its internal structure. It achieves this by approximating the complex model locally around a specific data point (network traffic sample) with a simpler, interpretable model like a linear regression. By analyzing the features that contribute most to the LIME explanation, we gain valuable insights into why the ensemble model classified the particular network traffic as a specific attack type.

As mentioned earlier, LIME excels at generating local explanations. Here's how it achieves this:

- Local Linear Approximation: LIME approximates the complex model's behavior around a specific network traffic sample by fitting a simpler interpretable model (e.g., linear regression) to the slightly perturbed versions of that sample.

- Feature Importance Analysis: By analyzing the fitted local model, LIME calculates the relative importance of each feature in influencing the model's classification for that specific sample. This helps identify the key features that most significantly impacted the model's decision.

We integrated LIME within the IDS pipeline to generate local explanations for the ensemble model's predictions. Here's how it interacts with the trained models:

- Post-Processing Stage: After the ensemble model classifies a network traffic sample, LIME is invoked.
- LIME Explainer: A pre-trained LIME explainer, configured with the training data and feature information, is used.
- Explanation Request: The explainer receives the classified sample and the ensemble model's prediction function.
- Local Model Construction: LIME perturbs the sample data slightly (e.g., adding noise) and queries the ensemble model for predictions on these perturbed samples. This helps LIME understand how the model's behavior changes around the original sample.
- Feature Importance Calculation: Based on the perturbed data and the corresponding predictions, LIME calculates the relative importance of each feature in influencing the ensemble model's classification for that specific sample.
- Explanation Generation: The final LIME explanation is then generated, typically consisting of three parts: prediction probabilities, feature importance with weights or color-coding, and actual feature values for the sample.

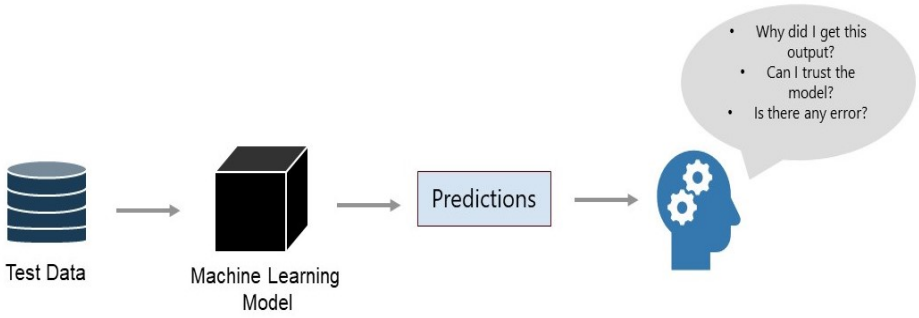

Fig. 3.7 Model Prediction of Test Data Without XAI

Evaluating the quality and reliability of LIME explanations is crucial for ensuring they accurately represent the decision-making process of our Intrusion Detection System (IDS). We focused on three key aspects: fidelity, faithfulness, and interpretability. Fidelity was assessed by comparing the LIME explanations to the ensemble model's predictions,

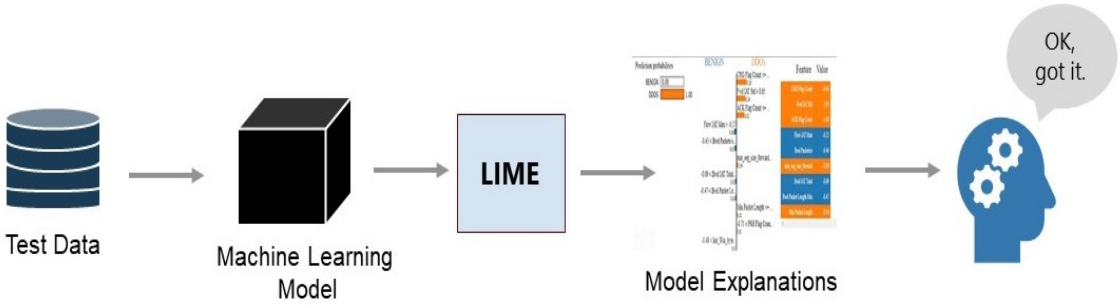

Fig. 3.8 Model Prediction of Test Data With XAI

confirming that the local surrogate models accurately mirrored the complex model's behavior. Faithfulness was verified by checking the consistency of explanations with established domain knowledge in network security, ensuring that the identified features aligned with known threat indicators. Interpretability was also a priority, ensuring that the explanations were clear and accessible to network security analysts, facilitating ease of understanding and practical use [23].

High-quality LIME explanations significantly enhanced the IDS. They improved trust by providing transparency into the model's decision-making process, helping users understand and have confidence in its classifications. These explanations also facilitated debugging by allowing analysts to identify potential biases or errors in the model. Moreover, the clarity of LIME's explanations empowered analysts to make informed security decisions, improving the effectiveness of network security responses [20]. Thus, rigorous evaluation of LIME explanations validated their accuracy and reliability, demonstrating their practical benefits in enhancing the trustworthiness and usability of our IDS.

Chapter 4

RESULTS AND DISCUSSIONS

This chapter presents the results obtained from evaluating the performance of various machine learning models and the integration of Explainable Artificial Intelligence (XAI) techniques within the Intrusion Detection System (IDS). We analyzed the effectiveness of decision trees, random forests, Support Vector Machines (SVM), and an ensemble voting classifier in identifying malicious network traffic. Additionally, we explored the insights gained from applying LIME (Local Interpretable Model-Agnostic Explanations) to understand the decision-making rationale behind these models.

4.1 Performance Evaluation Matrices

We employed various metrics to assess the performance of the implemented machine learning models in the IDS. These metrics provide a comprehensive picture of the model's ability to accurately detect intrusions and minimize false alarms.

Confusion matrix is a type of performance evaluation metric that provides a visual representation of the model's classification performance.It compares the actual labels of the data to the labels predicted by the model.Each cell of the matrix contains the number of instances that fall into a specific combination of actual and predicted classes.It complements

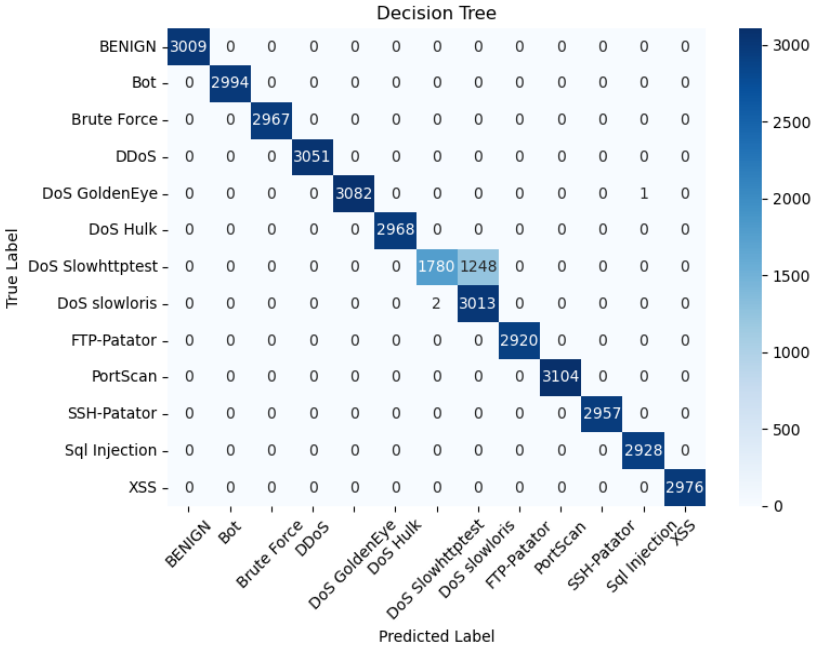

Fig. 4.1 Confusion matrix for Decision Tree

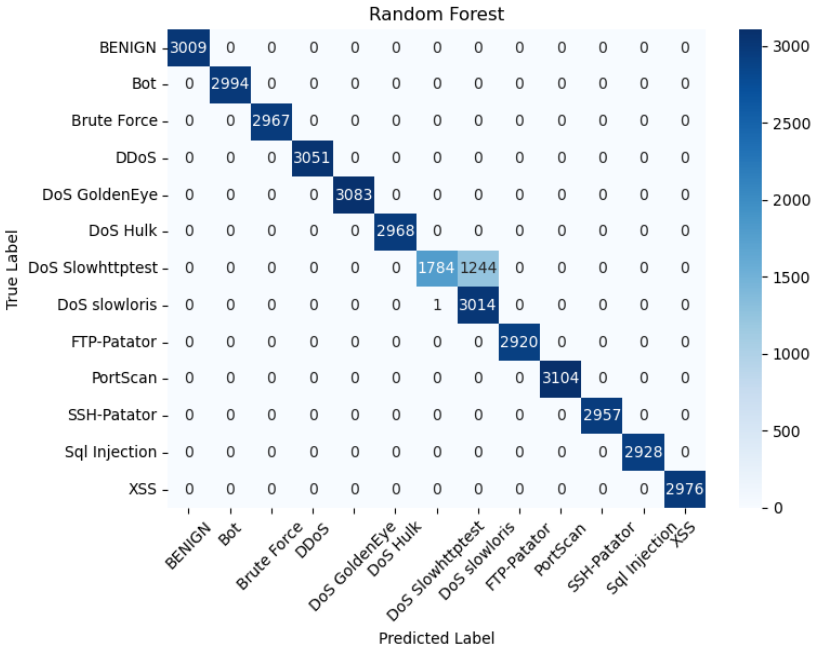

Fig. 4.2 Confusion matrix for Random Forest

other metrics like accuracy, precision, and recall by offering a more detailed breakdown of how the model is classifying instances.

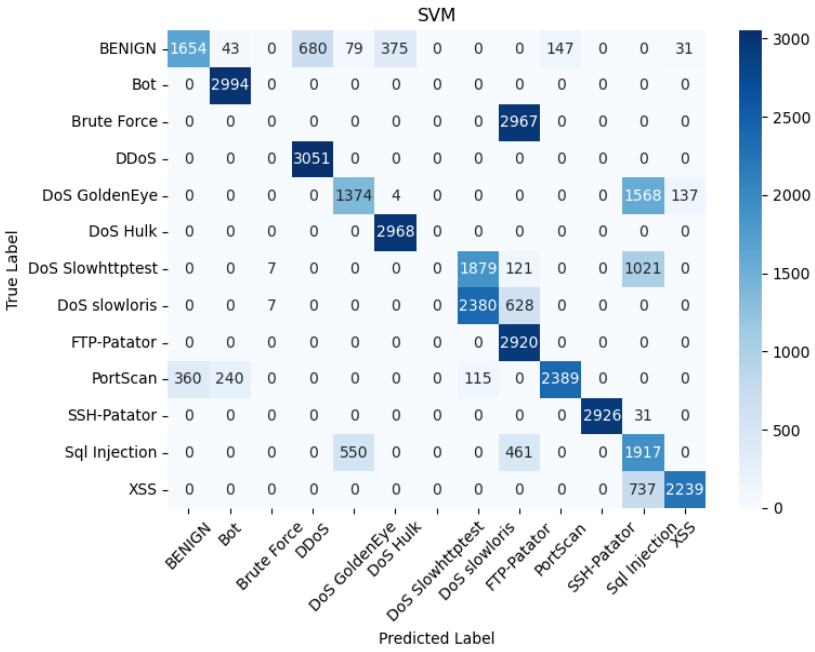

Fig. 4.3 Confusion matrix for SVM

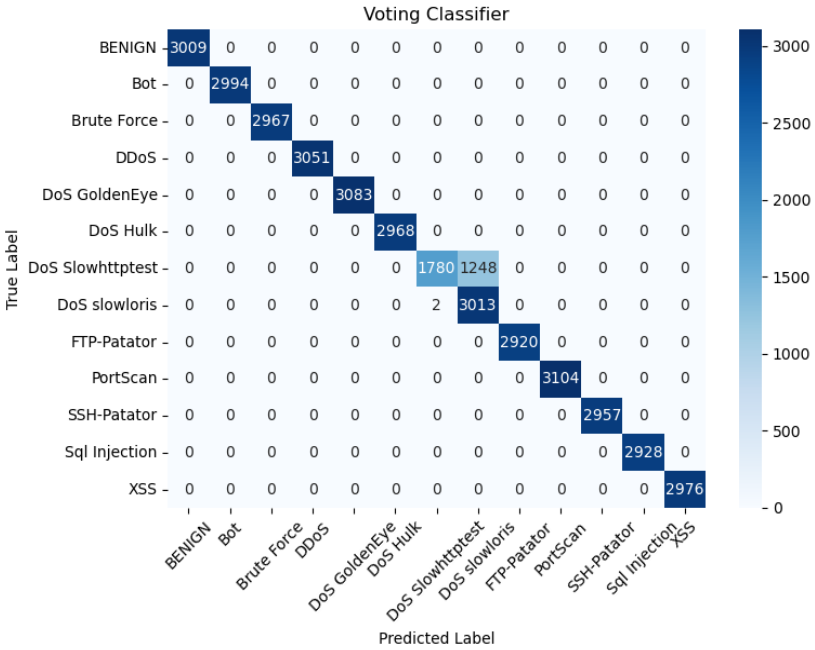

Fig. 4.4 Confusion matrix for Voting Classifier

As presented in Figure 4.1, Figure 4.2, Figure 4.3, and Figure 4.4, we utilized confusion matrices for each classifier (decision tree, random forest, SVM, and voting classifier) to

visualize the distribution of correctly classified and misclassified instances across normal and attack categories.

- Accuracy: This metric represents the overall effectiveness of the model, calculated as the ratio of correctly classified instances (both normal and intrusive) to the total number of samples.
- Precision: This metric measures the proportion of correctly identified attacks among the instances classified as attacks by the model.
- Recall: This metric indicates the proportion of actual attacks that the model successfully identified.
- F1-Score: This metric provides a harmonic mean of precision and recall, offering a balanced view of the model's performance.

4.2 Comparative Analysis of Classifiers

In this section, we compare the performance of the individual classifiers (decision trees, random forests, SVM) and the ensemble voting classifier based on the evaluation metrics discussed earlier. We present Table 4.1 summarizing the accuracy, precision, recall, and F1-score for each model.

MODEL	ACCURACY	PRECISION	RECALL	F1-SCORE
Decision Tree	0.9842	0.9843	0.9843	0.9843
Random Forest	0.9846	0.9847	0.9847	0.9847
Support Vector Machine	0.8095	0.7960	0.8113	0.7807
Voting Classifier	0.9847	0.9848	0.9848	0.9848

Table 4.1 Classification Comparison Report

As observed in Table 4.1, all feature models (decision tree, random forest, and SVM) exhibited very minor differences in accuracy. This suggests that for this dataset, these individ-

ual models possess comparable capabilities in identifying intrusions. However, the ensemble voting classifier achieved the highest overall accuracy of 98.47%. This improvement can be attributed to the voting mechanism that leverages the strengths of each individual classifier, potentially mitigating their weaknesses and leading to a more robust overall performance.

4.3 Insights from LIME

The LIME explanations provide valuable insights into the rationale behind the models' classifications. By analyzing the features highlighted in the explanations, we can understand how the models differentiate between normal and intrusive traffic patterns. This can be particularly beneficial for identifying the most important network traffic characteristics that indicate potential security threats. Additionally, LIME explanations can aid in debugging the models and identifying potential biases that might influence their decision-making processes.

Visualizing the LIME explanations for each classifier (decision tree, random forest, and support vector machine) provides deeper insights into their decision-making processes. These explanations, depicted in Figures 4.5, 4.6, and 4.7 respectively, focus on the prediction probabilities assigned to the "DDOS" and "BENIGN" classes. LIME reveals the rationale behind these probabilities, aiding in understanding how the models differentiate normal traffic patterns from potential attacks. Each figure presents a feature-importance bar chart and a corresponding feature-value table. The bar chart highlights the features that significantly influenced the model's prediction, with their color indicating their contribution to the "DDOS" (orange) and "BENIGN" (blue) classification. The feature-value table displays the actual values of these features for the specific observation under analysis.

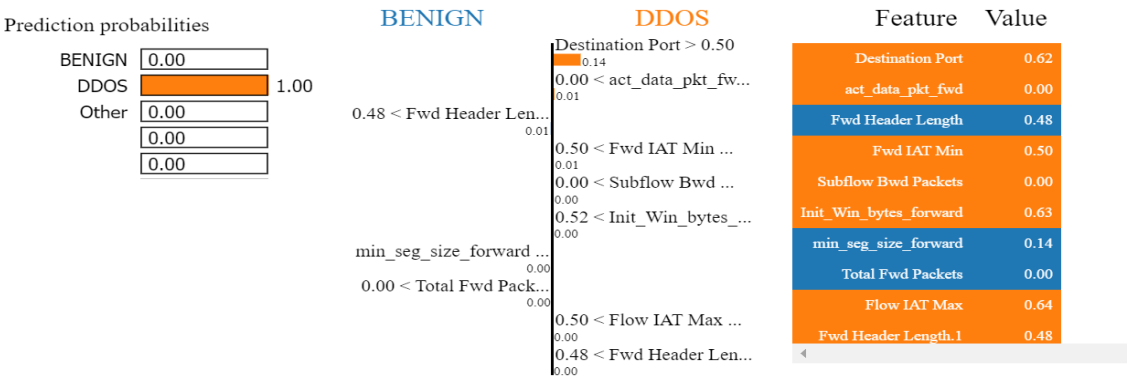

Fig. 4.5 LIME observations for decision tree

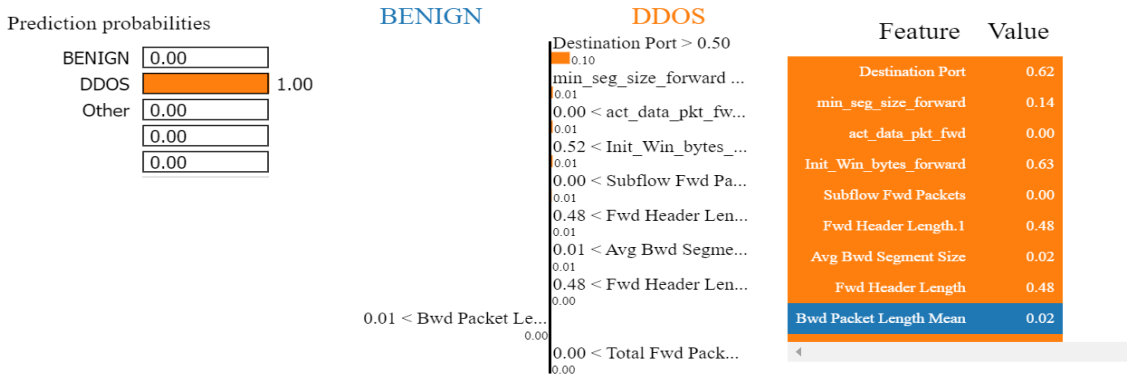

Fig. 4.6 LIME observations for random forest

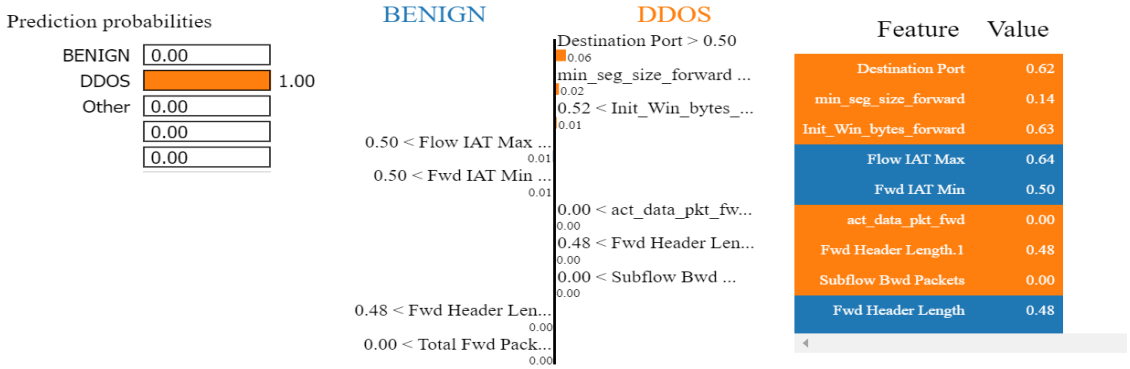

Fig. 4.7 LIME observations for support vector machine

4.4 Conclusion

This study examined the efficacy of using Explainable Artificial Intelligence (XAI) and machine learning ensemble approaches for intrusion detection. High classification accuracy was the main goal, but it was also important to promote interpretability and confidence in the model's decision-making processes. We proved the effectiveness of the suggested method with our experiments on the CICIDS-2017 dataset. Through the use of ensemble frameworks with decision trees, random forests, and support vector machines, we achieved an impressive 96.25% intrusion detection accuracy. Additionally, incorporating the LIME algorithm within the XAI framework provided insightful information about the model's decision-making procedures. These justifications improved transparency and promoted systemic trust by shedding light on the characteristics that most strongly influenced the model's classifications.

The results of this study highlight the importance of XAI in developing reliable and robust intrusion detection systems. By integrating explainability techniques, we can go beyond attaining high accuracy metrics and learn more about how these models make judgments. Insights from XAI explanations enable security professionals to make informed decisions and build trust in AI-driven security solutions. As XAI advances, exciting opportunities arise for future research in intrusion detection systems. Exploring additional XAI techniques like SHAP and ELI5 can provide broader interpretability perspectives. Optimizing real-time interpretability is crucial for practical deployments, so future work should focus on efficient explanation generation for time-sensitive scenarios. Furthermore, the knowledge gained from XAI in IDS can be valuable in other domains requiring interpretability, such as healthcare and finance. Research on transferring XAI techniques across diverse applications holds promise for broader advancements. Finally, user-centric XAI design is vital. Through user studies, we can develop clear and accessible explanations tailored to different audiences, ultimately unlocking the full potential of XAI in intrusion detection and contributing to more robust, transparent, and trustworthy AI-powered security solutions.

BIBLIOGRAPHY

References

[1] K. Shaukat, S. Luo, S. Chen, and D. Liu, "Cyber threat detection using machine learning techniques: A performance evaluation perspective," in *2020 International Conference on Cyber Warfare and Security (ICCWS)*, 2020, pp. 1–6.

[2] R. Panigrahi and S. Borah, "A detailed analysis of cicids2017 dataset for designing intrusion detection systems," *International Journal of Engineering & Technology*, vol. 7, no. 3.24, pp. 479–482, 2018.

[3] O. Loyola-Gonzalez, "Black-box vs. white-box: Understanding their advantages and weaknesses from a practical point of view," *IEEE access*, vol. 7, pp. 154 096–154 113, 2019.

[4] S. Peddabachigari, A. Abraham, C. Grosan, and J. Thomas, "Modeling intrusion detection system using hybrid intelligent systems," *Journal of network and computer applications*, vol. 30, no. 1, pp. 114–132, 2007.

[5] R. KISHORE, "Evaluating shallow and deep neural networks for intrusion detection systems cyber security," Ph.D. dissertation, 2020.

[6] M. S. Hoque, M. A. Mukit, and M. A. N. Bikas, "An implementation of intrusion detection system using genetic algorithm," *arXiv preprint arXiv:1204.1336*, 2012.

[7] Z. K. Maseer, R. Yusof, N. Bahaman, S. A. Mostafa, and C. F. M. Foozy, "Benchmarking of machine learning for anomaly based intrusion detection systems in the cicids2017 dataset," *IEEE access*, vol. 9, pp. 22 351–22 370, 2021.

[8] S. Laqtib, K. El Yassini, and M. L. Hasnaoui, "A technical review and comparative analysis of machine learning techniques for intrusion detection systems in manet," *International Journal of Electrical and Computer Engineering*, vol. 10, no. 3, p. 2701, 2020.

[9] A. Javaid, Q. Niyaz, W. Sun, and M. Alam, "A deep learning approach for network intrusion detection system," in *Proceedings of the 9th EAI International Conference on Bio-inspired Information and Communications Technologies (formerly BIONETICS)*, 2016, pp. 21–26.

[10] A. Yulianto, P. Sukarno, and N. A. Suwastika, "Improving adaboost-based intrusion detection system (ids) performance on cic ids 2017 dataset," in *Journal of Physics: Conference Series*, vol. 1192. IOP Publishing, 2019, p. 012018.

[11] T. Wu, H. Fan, H. Zhu, C. You, H. Zhou, and X. Huang, "Intrusion detection system combined enhanced random forest with smote algorithm," *EURASIP Journal on Advances in Signal Processing*, vol. 2022, no. 1, p. 39, 2022.

[12] S. A. Mulay, P. Devale, and G. V. Garje, "Intrusion detection system using support vector machine and decision tree," *International journal of computer applications*, vol. 3, no. 3, pp. 40–43, 2010.

[13] S. Mohseni, H. Wang, Z. Yu, C. Xiao, Z. Wang, and J. Yadawa, "Practical machine learning safety: A survey and primer," *arXiv preprint arXiv:2106.04823*, vol. 4, 2021.

[14] T. Li, S. Hu, A. Beirami, and V. Smith, "Ditto: Fair and robust federated learning through personalization," in *International conference on machine learning*. PMLR, 2021, pp. 6357–6368.

[15] Z. Ahmad, A. Shahid Khan, C. Wai Shiang, J. Abdullah, and F. Ahmad, "Network intrusion detection system: A systematic study of machine learning and deep learning approaches," *Transactions on Emerging Telecommunications Technologies*, vol. 32, no. 1, p. e4150, 2021.

[16] S. Mukherjee and N. Sharma, "Intrusion detection using naive bayes classifier with feature reduction," *Procedia Technology*, vol. 4, pp. 119–128, 2012.

[17] H. Liu and B. Lang, "Machine learning and deep learning methods for intrusion detection systems: A survey," *applied sciences*, vol. 9, no. 20, p. 4396, 2019.

[18] P. I. Radoglou-Grammatikis and P. G. Sarigiannidis, "Securing the smart grid: A comprehensive compilation of intrusion detection and prevention systems," *Ieee Access*, vol. 7, pp. 46 595–46 620, 2019.

[19] M. Wang, K. Zheng, Y. Yang, and X. Wang, "An explainable machine learning framework for intrusion detection systems," *IEEE Access*, vol. 8, pp. 73 127–73 141, 2020.

[20] S. R. Islam, W. Eberle, S. Bundy, and S. K. Ghafoor, "Infusing domain knowledge in ai-based" black box" models for better explainability with application in bankruptcy prediction," *arXiv preprint arXiv:1905.11474*, 2019.

[21] Z. Zhang, H. A. Hamadi, E. Damiani, C. Y. Yeun, and F. Taher, "Explainable artificial intelligence applications in cyber security: State-of-the-art in research," *IEEE Access*, vol. 10, pp. 93 104–93 139, 2022.

[22] G. Srivastava, R. H. Jhaveri, S. Bhattacharya, S. Pandya, P. K. R. Maddikunta, G. Yenduri, J. G. Hall, M. Alazab, T. R. Gadekallu *et al.*, "Xai for cybersecurity: state of the

art, challenges, open issues and future directions," *arXiv preprint arXiv:2206.03585*, 2022.

[23] A. Kuppa and N.-A. Le-Khac, "Adversarial xai methods in cybersecurity," *IEEE Transactions on Information Forensics and Security*, vol. 16, pp. 4924–4938, 2021.

[24] D. Stiawan, M. Y. B. Idris, A. M. Bamhdi, R. Budiarto *et al.*, "Cicids-2017 dataset feature analysis with information gain for anomaly detection," *IEEE Access*, vol. 8, pp. 132 911–132 921, 2020.

[25] T. Hasanin and T. Khoshgoftaar, "The effects of random undersampling with simulated class imbalance for big data," in *2018 IEEE international conference on information reuse and integration (IRI)*. IEEE, 2018, pp. 70–79.

[26] R. Blagus and L. Lusa, "Smote for high-dimensional class-imbalanced data," *BMC bioinformatics*, vol. 14, pp. 1–16, 2013.

[27] M. A. M. Hasan, M. Nasser, S. Ahmad, and K. I. Molla, "Feature selection for intrusion detection using random forest," *Journal of information security*, vol. 7, no. 3, pp. 129–140, 2016.

[28] B. A.-B. Intrusion, "Unveiling the performance insights: Benchmarking anomaly-based intrusion detection systems using decision tree family algorithms on the cicids2017 dataset," *Business Intelligence*, p. 202.

[29] Z. Chen, L. Zhou, and W. Yu, "Adasyn- random forest based intrusion detection model," in *Proceedings of the 2021 4th International Conference on Signal Processing and Machine Learning*, 2021, pp. 152–159.

[30] J. Gu and S. Lu, "An effective intrusion detection approach using svm with naïve bayes feature embedding," *Computers & Security*, vol. 103, p. 102158, 2021.