I.J. Wireless and Microwave Technologies, **2026, 1, 50-62**
Published Online on February 8, 2026 by MECS Press (http://www.mecs-press.org/)
DOI: 10.5815/ijwmt.2026.01.04

Enhancing Intrusion Detection for Minority Attack Classes: A SHAP-Based Feature Selection Approach with Deep Neural Networks

Anagha A. S.*
College of Engineering Trivandrum, aff. to APJ Abdul Kalam Technological University, Trivandrum, 695016, India
E-mail: anaghasony10@gmail.com
ORCID iD: https://orcid.org/0000-0002-3857-1875
*Corresponding Author

Ciza Thomas
Digital University Kerala, Trivandrum, 695317, Kerala, India
E-mail: cizathomas@gmail.com
ORCID iD: https://orcid.org/0000-0002-1030-3000

Sreelatha G.
College of Engineering Trivandrum, Trivandrum, 695016, India
E-mail: sreelathag@cet.ac.in
ORCID iD: https://orcid.org/0000-0002-7807-0119

Received: 07 May, 2025; Revised: 20 July, 2025; Accepted: 14 September, 2025; Published: 08 February, 2026

Abstract: In the dynamic landscape of cybersecurity, safeguarding computer networks against persistent malicious threats is paramount. Intrusion Detection Systems are crucial in this context by monitoring network traffic for unauthorized access. While the integration of Machine Learning and Deep Learning has significantly advanced intrusion detection, the persistent challenge lies in effectively detecting minority attack classes. This study introduces an innovative approach that combines SHapley Additive exPlanations(SHAP) for feature selection and Deep Neural Networks(DNN) to enhance the performance of intrusion detection systems, particularly focusing on minority attack classes in the NSL-KDD dataset. Applied to a Random Forest classifier using a balanced dataset, SHAP provides valuable insights into feature importance, refining the feature set for seamless integration into a DNN architecture. Employing the NSL-KDD dataset, the research concentrates on elevating the detection accuracy for User-to-Root attack and Root-to-Local attacks. The results showcase a notable improvement in performance along with a reduction in computational time compared to using all the available features. A key emphasis of the study is on detecting all attack types without compromising the F1-score. An in-depth analysis of the initial set of 41 features identifies 30 as crucial for effective intrusion detection. On the imbalanced dataset, SHAP-based feature reduction improved the overall F1-score in multiclass classification from 86% to 91% by reducing training time by 8.86%, confirming that SHAP can lower complexity without sacrificing accuracy. However, several minority attacks remained undetected due to their extremely low representation. Additional experiments with oversampled data confirm that SHAP continues to provide efficiency gains while enabling robust detection of rare attack classes. These findings demonstrate that SHAP-based feature selection improves efficiency in IDS and has strong potential for minority attack detection if data scarcity is addressed. This research not only contributes to the enhancement of IDS capabilities but also highlights the importance of meticulous feature selection in achieving comprehensive and efficient intrusion detection.

Index Terms: Intrusion Detection Systems, Random Forest, Deep Neural Network, SHAP, User-to-root attack, Root-to-Local attack

1. Introduction

In the contemporary digital era, prioritizing the computer networks and systems security is absolutely crucial due to the persistent threat posed by various malicious activities. With an ever-increasing reliance on digital infrastructure, the vulnerabilities within these networks make them susceptible to a multitude of security risks. In

response to these challenges, Intrusion Detection Systems (IDS)[1] have emerged as a critical barrier, holding a key position in safeguarding the integrity and confidentiality of information.

The main function of Intrusion Detection Systems is to diligently inspect network flow, actively seeking signs of unauthorized or suspicious access that could indicate potential security breaches. By employing sophisticated algorithms and rule sets, IDS aims to identify patterns indicative of malicious activities, providing an early warning system to network administrators. The efficacy of an IDS is crucially measured through its capability to accurately and promptly detect intrusions, achieving equilibrium between sensitivity and specificity to minimize false alarms.

Considering the fluid nature of cyber threats, researchers and experts continually explore innovative techniques to elevate the capabilities of IDS. This pursuit reflects the recognition that the conventional methods of intrusion detection may not suffice in addressing the advancing methodologies of cyber adversaries. The research community engages in the evolution of sophisticated algorithms, machine learning models, and data analysis methodologies to fortify IDS against sophisticated and emerging threats.

Innovations in intrusion detection extend beyond merely identifying known attack signatures. Researchers delve into anomaly detection, behavior analysis, and predictive modeling to detect previously unseen threats. Machine learning and artificial intelligence play a pivotal role in this evolution, enabling IDS to adjust and glean insights from fresh data patterns, thereby enhancing their capacity to identify novel and advanced intrusions.

Moreover, the continuous exploration of innovative techniques in intrusion detection aligns with the broader cyber-security strategy of proactive defense. Instead of merely responding to known threats, the focus is on anticipating and mitigating potential risks, thereby fortifying the resilience of computer networks against a diverse array of cyber threats.

The competencies of various Intrusion Detection Systems [2] have undergone significant advancements, particularly with the integration of Machine Learning and Deep Learning techniques. However, a persistent challenge remains in the accurate detection of less common attack categories of intrusions that are relatively rare in real-world scenarios. Recognizing the crucial importance of tackling this gap, this research adopts a novel approach to enhance Intrusion Detection Systems, specifically concentrating on improving their efficacy in detecting minority attack classes.

The fundamental challenge lies in the inherent rarity of minority attack classes, which poses a unique difficulty for IDS in distinguishing these infrequent instances from normal network behavior. Traditional approaches often struggle to accurately identify minority attacks due to the scarcity of relevant data for training and modeling. As a result,there is a critical necessity to devise more efficient methodologies to bolster the detection capabilities specifically tailored for these less frequent but potentially more sophisticated and impactful attacks.

In response to the identified challenge, this research work introduces a pioneering fusion of techniques, combining SHAP (SHapley Additive exPlanations) for feature selection with Deep Neural Networks (DNNs) for classification [3]. SHAP [4], based on the principles of game theory, provides a robust platform for determining the importance of individual features in predictive models. By applying SHAP for feature selection, the research seeks to improve the discriminative power of the IDS by focusing on the features highly pertinent to the detection of minority attack classes.

The integration of Deep Neural Networks into this framework capitalizes on the strengths of DNNs in exploring intri-cate patterns and representations within datasets through learning. The deep learning framework is particularly well-suited for capturing complex relationships and nuances within the dataset, making it a potent tool for handling the intricacies associated with minority attack classes. The fusion of SHAP for feature selection with DNNs for classification creates a synergistic approach that leverages the interpretability of SHAP and the modeling capabilities of DNNs. By combining these techniques, the research aims to achieve a twofold enhancement: first, by identifying and prioritizing the features crucial for detecting minority attack classes, and second, by leveraging the power of DNNs to effectively classify and distinguish these less frequent instances from normal network behavior. The ultimate target is to significantly enrich the performance and trustworthiness of Intrusion Detection Systems, particularly in scenarios where the rarity of certain attacks poses a formidable challenge.

SHAP (SHapley Additive exPlanations) [5] stands out as an advanced and influential methodology in the realm of explainable AI. It renders a systematic and quantitative way to grasp the influence of individual features on the forecasts generated by machine learning models. In the context of intrusion detection, the application of SHAP to a Random Forest (RF) classifier with a balanced dataset is particularly insightful. When SHAP is applied to an RF classifier operating on a balanced dataset, it provides a holistic understanding of feature importance. This understanding is crucial for gaining insights into the factors that drive intrusion detection decisions. By quantifying the effect of individual feature, SHAP enhances the interpretability of the model, offering a nuanced view of the discriminative power of individual features in categorizing internet traffic as either normal or suggestive of an intrusion. This process of SHAP-based feature evaluation results in a refined feature set, which is enriched with valuable insights obtained through the SHAP methodology [4].

The NSL-KDD dataset [6], which includes five target classes, acts as the experimental basis for applying SHAP and subsequent selection of features. The chosen dataset, with its diverse set of network traffic instances, allows for a comprehensive evaluation of the proposed methodology across various intrusion types. The refined feature set obtained through SHAP [7] is then employed in the structure of a Deep Neural Network. Leveraging the insights gained from SHAP's feature selection, the DNN is equipped with a more informed and discriminative set of features. This integration boosts the DNN's capability to recognize intricate patterns and relationships within the datasets, contributing to enhanced performance in intrusion detection.

A noteworthy outcome of this research is the enhanced performance of the DNN [8], especially in the detection of minority attack classes such as U2R(User to Root) [6] and R2L(Remote to Local). The refined feature set, curated with

the assistance of SHAP, proves particularly effective in addressing the challenges posed by less frequent attack categories. This nuanced approach not only enhances the accuracy of detecting minority attacks but also demonstrates a reduction in computational time compared to using the entire set of features. The efficiency gains signify the practical utility of SHAP-based feature selection in optimizing the computational resources required for intrusion detection tasks.

The application of SHAP to intrusion detection, specifically in the context of a balanced RF classifier and subsequent feature selection for a DNN, represents an approach to improving the interpretability and performance of intrusion de-tection systems. The research underscores the significance of understanding feature importance, especially when dealing with minority attack classes, and highlights the practical benefits of SHAP in refining feature sets for more efficient and accurate detection of intrusion.

The remaining sections of the paper are structured as follows. Section 2 reviews the related works, while Section 3 gives a brief depiction of the background. Section 4 elaborates on the methodology, while Section 5 delves into the results and limitations. The conclusion and future scope are then outlined in Section 6.

2. Related Work

The field of cybersecurity has been the focus of extensive exploration, with particular emphasis on elevating the effectiveness of IDS. While the development of numerous IDS solutions has demonstrated their efficacy, a persistent challenge looms in the form of understanding why certain attacks manage to evade detection by various IDSs. This challenge becomes particularly pronounced when incorporating feature selection techniques into the IDS framework, as there is a substantial risk of inadvertently omitting critical features crucial for the detection of specific types of attacks.

The crux of this challenge manifests prominently when dealing with less frequent attack categories, notably R2L (Remote-to-Local) and U2R (User-to-Root) attacks [6]. These attack types occur less frequently compared to normal network traffic, making them inherently harder to detect. Additionally, attacks falling under the umbrella of probe attacks further contribute to the complexity of the issue. These less frequent attack categories pose a unique challenge due to the scarcity of relevant instances in the dataset, creating an imbalance in the data distribution.

Addressing this data imbalance is imperative for achieving accurate and robust IDS performance. One potential avenue is the implementation of data balancing methods, which involve techniques to rectify the disproportionate representation of different attack classes. By artificially increasing the number of instances belonging to minority attack classes before engaging in feature selection, we can mitigate the risk of overlooking critical features associated with less frequent attacks. In the context of R2L, U2R, and probe attacks, this proactive approach to data balancing holds significant promise.

By augmenting the instances of these minority attack classes, we ensure that feature selection techniques have a more comprehensive representation of the entire attack landscape. This not only contributes to overcoming the challenge posed by less frequent attack categories but also reinforces the resilience and accuracy of IDSs admist evolving cybersecurity threats.

Thus, the integration of data balancing methods, coupled with a strategic increase in the number of minority attack class instances, serves as a proactive measure to address the inherent challenges associated with feature selection in IDS. This approach fosters a more inclusive understanding of the attack landscape, ultimately bolstering the capability of IDSs to detect and mitigate a broader spectrum of cyber threats.

Thabtah et al. [9] contribute valuable insights into addressing data imbalance issues in the context of the Naive Bayes classifier. Their work delves into various methodologies such as sampling techniques, cost-sensitive learning, and hybrid approaches. The overarching goal is to rectify imbalances in the dataset, ultimately enhancing the detection rate of minority data. This comprehensive exploration of strategies demonstrates the versatility of the Naive Bayes classifier and underscores the importance of tailored approaches to mitigate data imbalance challenges.

Krawczyk's paper [10] delves into the broader landscape of data imbalance, providing a comprehensive overview of the myriad issues and challenges associated with this prevalent problem. By elucidating the complexities of imbalanced datasets, Krawczyk sets the stage for a deeper comprehension of the complexities associated with managing such data. This foundational knowledge is crucial for devising effective solutions to address data imbalance issues across various machine learning applications.

In the realm of feature selection, Mukkamala et al.[11] present a method to rank input features, emphasizing its sig-nificance in enhancing the performance of classifiers. Their work focuses on evaluating the influence of chosen features on two prominent classifiers, namely Artificial Neural Networks (ANN) and Support Vector Machines (SVM) [12]. The findings of their study reveal that SVM outperforms ANN in terms of both training time and running time. This compara-tive analysis offers valuable perspectives on the strengths and weaknesses inherent in various classifiers, abetting experts and researchers in making informed choices based on their specific requirements and constraints.

Collectively, these works offer to the ongoing discussion on addressing data imbalance in machine learning. Thabtah et al.'s [9] exploration of methodologies specific to the Naive Bayes classifier, Krawczyk's [10] comprehensive examination of data imbalance challenges, and Mukkamala et al.'s [11] evaluation of feature selection impact on classifiers collectively enrich the knowledge base in the pursuit of robust and effective solutions for handling imbalanced datasets in various machine learning applications.

In the research carried out by Lundberg et al. [13], the focus is on enhancing the interpretability of tree-based algorithms through the introduction of a novel method known as SHAP (SHapley Additive exPlanations). The main goal of this method is to calculate the significance of features in predicting output target classes. SHAP achieves this by leveraging Shapley values from game theory, providing a rigorous and theoretically grounded framework for attributing feature importance. This work represents a significant advancement in understanding and interpreting the processes of making decisions of tree-based algorithms, thereby contributing to the broader goal of improving model transparency and interpretability.

Building on the foundation laid by Lundberg et al., Alenezi et al. [7] further explores the application of SHAP explanation methods in determining important features across various machine learning models, including Random Forest and XGBOOST. The study specifically delves into different SHAP explanations, such as tree-based SHAP and kernel-based SHAP, showcasing the versatility of SHAP in diverse model architectures. By extracting important features through various SHAP techniques, Alenezi et al. contribute to the understanding of feature contributions in different machine learning paradigms, fostering a more comprehensive approach to model explainability.

In the work presented by Perez et al. [14], a meticulous assessment of the SHAP methodology is conducted. This study goes beyond the conventional use of SHAP and explores a modified approach tailored for precisely calculating Shapley values, specifically catering to decision tree methods. The researchers provide a detailed comparative analysis, pitting this modified method against the model-agnostic SHAP technique. This comparative study is situated in the context of compound activity and potency value predictions, illuminating the strengths and drawbacks of different SHAP methodologies. Rodriguez Perez et al.'s work adds a nuanced layer to the existing understanding of SHAP and its applicability in various scenarios, thereby contributing to the ongoing efforts to refine and tailor SHAP methods to various kinds of machine learning models.

Collectively, the inquiry of Lundberg et al., Alenezi et al., and Perez et al. represent a continuum in the exploration and enhancement of SHAP explanation methods. From Lundberg et al.'s foundational work on introducing SHAP to improve tree-based algorithm interpretability, to Alenezi et al.'s broadening of its application across diverse machine learning models, and finally to Rodriguez Perez et al.'s meticulous assessment and modification of SHAP for decision tree methods, these works collectively advance our understanding of model explainability and the nuanced interpretation of machine learning models.

Recent works have demonstrated the limitations of classical feature selection methods such as Information Gain, Mu-tual Information, and Recursive Feature Elimination, which often fail to capture feature interactions and provide limited interpretability. Marcilio and Eler in their work [15] showed that SHAP consistently outperformed these approaches when integrated with XGBoost, offering both higher predictive performance and transparent explanations of feature contribu-tions. The study [16] by Anagha et al. in, SHAP-based feature selection has further been shown to improve F1-scores on NSL-KDD compared to correlation-based and information gain methods, while reducing computational costs. These findings support the adoption of SHAP as a robust and interpretable feature selection method.

In the study [17] conducted by Vinayakumar et al. , the researchers propose a deep neural network (DNN) framework designed for the detection of both Host-based Intrusion Detection Systems (HIDS) and Network-based Intrusion Detection Systems (NIDS). The key highlight of their work is the demonstrated superiority of the DNN framework over previously implemented classical Machine Learning (ML) classifiers in the realms of both HIDS and NIDS. This finding suggests that the DNN framework exhibits enhanced performance and predictive capabilities, marking a significant advancement in the field of intrusion detection.

However, a notable limitation of the proposed DNN framework is identified in the study that it does not provide detailed information about the structure and characteristics of the malware. This aspect is crucial for understanding the specific attributes and behaviors of malicious entities, which is essential for developing effective countermeasures and refining intrusion detection strategies. The absence of detailed information about the malware's structure and characteristics may hinder the interpretability of the DNN framework's decisions. In practical terms, while the DNN framework excels in detecting intrusions, the lack of insights into the underlying features and patterns of the malware limits the ability to perform in-depth analysis and develop targeted responses. Understanding the structural and behavioral characteristics of malware is instrumental in refining and adapting intrusion detection systems to evolving threats.

To address this limitation, future research in this domain could focus on incorporating additional layers or modules within the DNN framework to extract and elucidate the structural attributes of detected malware. This could involve integrating explainability techniques, such as feature attribution methods, to highlight the specific features contributing to the classification decisions made by the DNN. This work presents an innovative DNN framework that outperforms classical ML classifiers in intrusion detection. However, the lack of detailed information about the malware's structure and characteristics represents a potential avenue for further research, emphasizing the importance of not only detecting intrusions but also gaining insights into the nature of the detected malware for more effective cybersecurity measures.

3. Background

3.1. Random forest classifier

Random Forest (RF) [18] is a robust and versatile ensemble learning technique widely utilized within the domain of machine learning and data analytics. Its popularity stems from its ability to significantly enhance predictive accuracy across various domains and effectively handle diverse types of data. The fundamental idea behind Random Forest lies in its ensemble nature, where it combines the predictions of multiple decision trees to produce more reliable and accurate results. While decision trees [19] are susceptible to overfitting and may not generalize well to unseen data, Random Forest mitigates these issues through its ensemble approach.

Random Forest is particularly skilled in managing high-dimensional datasets. In scenarios where the attributes num-ber is large, and some attributes may be irrelevant or redundant, RF can effectively filter out noise and focus on the most informative attributes. This adaptability to high-dimensional data makes it well-suited for applications in detecting net-work attacks, where datasets frequently comprise a multitude of features.In this work, overfitting issues resulting from oversampling are mitigated by employing 5-fold cross-validation with a Random Forest classifier using 100 estimators.

3.2. Random oversampling

Random Oversampling [9] is a strategic technique employed in the realm of machine learning to tackle the challenge of class imbalance within a dataset. Class imbalance [10] arises when there is a substantial under representation of one class,often the minority class , in comparison to the more represented class. occurs when one class, usually the minority class, is significantly underrepresented compared to the majority class. This situation is common in the field of cyber security, where the occurrence of the attack is infrequent compared to normal traffic, and also some attacks are rare compared to some other attacks.

The goal of Random Oversampling is to rectify this imbalance by artificially increasing the number of instances belonging to the minority class. This process involves the random duplication or replication of existing samples from the minority class until a desired balance is achieved between the minority and majority classes. The underlying idea is to provide the machine learning model with a more equitable representation of both classes, allowing it to learn patterns associated with the minority class more effectively.

By arbitrarily replicating instances from the less represented attack class, Random Oversampling introduces diversity into the training data for the model. This diversity aids in averting the model from becoming excessively skewed towards the majority class, which could lead to poor generalization and suboptimal performance on the minority class. Addition-ally, it can be particularly beneficial when the available data for the minority class is limited, and the model struggles to discern meaningful patterns due to the scarcity of samples in the training dataset.

However, while Random Oversampling can be effective in mitigating class imbalance, it comes with certain con-siderations and potential drawbacks. One concern is the risk of introducing noise into the training data, especially if the minority class instances are duplicated excessively. This can lead to the model memorizing specific instances rather than learning true underlying patterns. Careful consideration of the degree of oversampling and its impact on model per-formance is important to strike an equilibrium between addressing class imbalance and maintaining the integrity of the learning process.

3.3. SHapley Additive exPlanations

SHapley Additive exPlanations (SHAP) [4] stands out as a powerful and interpretable machine learning technique designed to illuminate the internal mechanisms of intricate models. Its primary purpose is to offer comprehensive eluci-dations on the predictions generated by machine learning models, particularly those that might be considered black-box models due to their intricate and challenging-to-understand nature. The fundamental concept behind SHAP is rooted in Shapley values, a concept originating from cooperative game theory.

Cooperative game theory is concerned with the fair distribution of a value among a group of players who collabora-tively contribute to achieving a certain outcome. Shapley values, within this context, quantify the marginal contribution of each player to every possible combination of players in the coalition. This fairness principle ensures that each player is rewarded in proportion to their individual impact on the group's performance.

Within the framework of machine learning, SHAP leverages Shapley values to attribute the prediction of a model to each feature in a fair and consistent manner. It achieves this by examining every possible combinations of attributes and determining the average contribution of each attributes to the prediction of model. This process allows for a nuanced understanding of how individual features influence the outcome of a prediction, providing valuable perspectives into the model's decision-making mechanism.

One of the significant advantages of SHAP is its applicability to a wide range of machine learning models, including complex ensemble models and deep neural networks. This versatility makes it a valuable tool for model interpretation across various domains. By quantifying the impact of each feature through SHAP values, users can gain a more trans-parent and granular understanding of the factors that contribute to a particular prediction, facilitating model debugging, validation, and trust-building in real-world applications.

Furthermore, SHAP values not only provide insights into the direction and extent of a feature's impact but also provide a means to visualize and interpret [20] these contributions. Plots [13] such as SHAP summary plots and force plots enable users to grasp the importance and directionality of each feature for individual predictions, enhancing the interpretability of complex models.

3.4. Deep Neural Networks

Deep Neural Networks (DNNs) [17] represent a subset of artificial neural networks characterized by multiple layers of interconnected nodes or neurons. In the context provided, a deep neural network [3] is specified with three layers: an input layer, a hidden layer, and an output layer. The architecture described follows a feedforward design, where information is propagated in a unidirectional manner from the input layer through the hidden layer to the output layer. This forward propagation occurs with full connectivity between neurons in each layer. The activation functions used in the network play a crucial role in introducing non-linearity, allowing the model to learn complex patterns and relationships within the data. In the hidden layer, the Rectified Linear Unit (ReLU) [21] activation function is employed . ReLU is preferred for its ability to mitigate issues like vanishing gradients, which can impede the training of deep networks. The vanishing gradient issue arises when the gradients become diminutive during backpropagation, impeding the adjustment of weights and thus slowing down the learning process.

The output layer, designed for multiclass classification, utilizes the softmax activation function. Softmax [22] converts the network's initial output into probabality distributions across numerous classes so is frequently utilized in the concluding layer of a neural network for multiclass issues. This enables the model to assign a probability to each class, facilitating the selection of the most likely class for a given input.

The hidden layers are described as dense layers with 12 neurons each. The choice of the number of neurons in the hidden layers is often determined through experimentation and tuning. The 12 neurons per hidden layer represent a hyperparameter that influences the capacity and complexity of the model. Adjusting the number of neurons can impact the network's ability to learn intricate patterns in the data and may also help prevent overfitting. This compact design was chosen to reduce computational cost while allowing iterative evaluation with different feature subsets.

The input layer, as specified, consists of 122 neurons for the baseline model without feature selection. However, it is noted that the number of input neurons may vary when feature selection is applied. Feature selection involves choosing a subset of relevant features from the original set, potentially reducing the dimensionality of the input and focusing on the most informative attributes.

For model training, the prediction loss is calculated using the Categorical Cross Entropy, a common choice for issues in multiclass classification. This loss function evaluates the discrepancy between the predicted probability distribution and the actual distribution of class labels. The model is optimized using the Adam optimizer,a widely used optimization algorithm that adjusts the learning rates for each parameter dynamically during training, helping to converge faster and more efficiently.

To avoid overfitting, early stopping, dropout, and batch normalization were applied. Data was split 70/30 into training and testing, a common practice in IDS studies. While cross-validation was not employed due to computational overhead, robustness was supported through these regularization measures. Model performance was reported using precision, recall, and F1-scores, including both macro and weighted averages.

3.5. Evaluation Metrics

The model is evaluated using performance parameter F1-score ([21]), which gives the relation between Precision and Recall.

- Precision: Precision is the ratio of data instances predicted as positive (True positives TP) that are actually positive ([23]).

$$Precision, P = \frac{TP}{TP+FP} \tag{1}$$

- Recall: It is the measure of fraction of correct positive predictions made out of all positive predictions([23]).
- F1-score: The F1-score is harmonic mean of precision and recall, to represent both in one metric([23]).

$$Recall = \frac{TP}{TP+FN} \tag{2}$$

$$F1 - Score = \frac{2 * (\text{Precision} * \text{Recall})}{\text{Precision} + \text{Recall}} \tag{3}$$

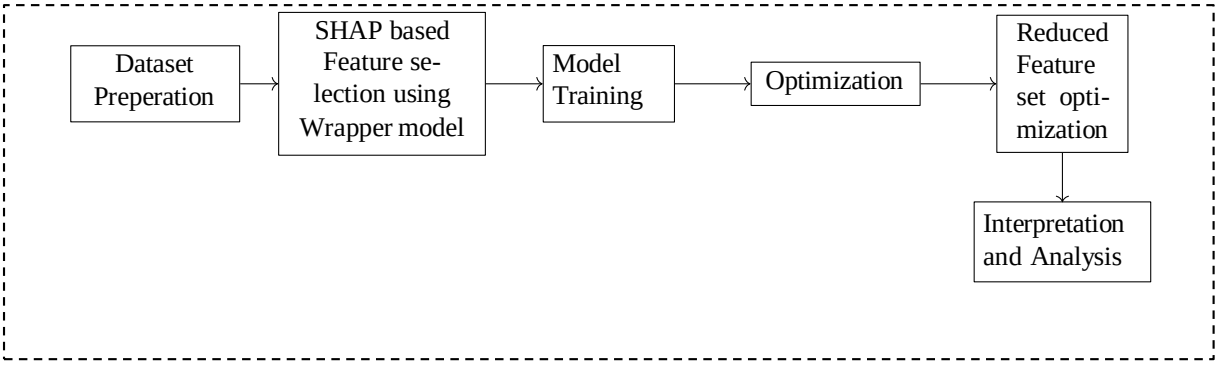

Fig. 1. Methodology of the proposed method

In addition to reporting scores for each attack class, two summary measures were considered: macro-averaged F1-score, which treats all classes equally and highlights performance on minority categories, and weighted F1-score, which adjusts results according to class distribution to reflect overall performance more realistically.

4. Methodology

This comprehensive approach is designed to elevate the capabilities of Intrusion Detection Systems (IDS) by seamlessly integrating two potent technologies: SHAP (SHapley Additive exPlanations) for feature selection and Deep Neural Networks (DNNs) for intrusion detection. The pivotal role of SHAP values in gauging the impact of each feature on specific predictions is harnessed to precision within this approach. By leveraging SHAP, the methodology adeptly identifies and selects the most influential features, fostering a refined and targeted representation of the data. DNNs, with their adeptness at handling high-dimensional data often encountered in intrusion detection scenarios, autonomously acquire hierarchical features.

This intrinsic capability empowers the model to discern subtle patterns indicative of network intrusions with unparal-leled accuracy. The primary goal is to augment the precision of IDS, with a specific focus on fortifying its capabilities in detecting minority-class attacks. Through the harmonious fusion of SHAP's feature selection acumen and DNNs' profi-ciency in deciphering intricate patterns, this approach envisions a heightened level of resilience and accuracy in the realm of intrusion detection, ensuring robust protection against a spectrum of cyber threats.

Figure 1 shows the process diagram of the proposed method. The various steps for the proposed methods are as follows:

1. Pre-processing of data involves applying various techniques to prepare raw data for machine learning. This can include stages such as cleaning, normalization, handling missing values, categorical variable encoding, feature scaling and data balancing. The dataset is then split into 70% and 30% for training and testing respectively. For training the model we use the training data and for assessing the model's performance on unseen examples the testing data is employed.
2. Selection of features is performed using the SHAP explanation method. The SHAP explanation of an RF classifier output with oversampling is utilized to identify the significant features in predicting the output.
3. Train the model and throughout the training stage, the model discerns patterns and correlations in the data, refining its parameters to enhance prediction accuracy.
4. Optimize the model's hyperparameters and assess its performance on the test data using the evaluation metrics like F1-score.
5. Evaluate the F1-score for each target class and incorporate new features to the reduced feature set based on the importance of each feature in the SHAP explanation if the result is low. Then, repeat steps 3 and 4 until satisfac-tory results are achieved for minority attack classes.The optimal model for achieving the highest F1-score for the minority attack classes has been constructed.
6. Compare the reduced feature set model's performance with the model trained on all features to validate the benefits of feature reduction.The model's capacity to make precise predictions on new, unseen data is a crucial measure of its effectiveness.

5. Results and Analysis

All experiments were conducted in Python 3.8.20 on a machine equipped with an Intel i7 processor and 16 GB RAM. The deep learning models were implemented in TensorFlow 2.11.0 with Keras 2.11.0, while Random Forest and evaluation metrics were implemented using scikit-learn 1.3.2. SHAP explanations were generated using the SHAP library 0.44.1, and random oversampling was performed using the imbalanced-learn library 0.12.4. A fixed random seed (42) was applied to RF but for DNN experiments, no fixed seed were used, but repeated runs confirmed that the performance

differences were consistent. The NSL-KDD dataset is utilized. The model is assessed using the F1-score performance parameter, offering a balance between precision and recall, making it especially valuable in cases of class imbalance.

The NSL-KDD dataset was selected because it offers a consistent and reproducible testbed for evaluating SHAP-based feature selection. This choice made it possible to clearly demonstrate the effectiveness of the approach, but future validation on more recent datasets will be important to confirm its relevance in real-world scenarios.

The NSL-KDD dataset [6] utilized in this study is a modified iteration of the KDD cup 99 dataset. It has a total of 125973 data samples out of which 70% is used for training and 30% for testing. The NSL-KDD dataset contains five target classes, including four distinct attack classes and one normal class. The four attack classes are DoS (Denial of Service), Probe, R2L (Remote to Local), and U2R (User to Root), with R2L and U2R representing the minority attack classes.

This study focuses on U2R and R2L minority attack classes. The NSLKDD dataset contains buffer-overflow, loadmod-ule, perl, and root-kit as examples of U2R attacks, and ftp-write, guess-password, imap, multihop, phf, spy,warezclient, and warezmaster as examples of R2L attacks.

Table 1. Important Features/Attributes in the descending order after SHAP explanation on RF classifier

Sl.No.	Attributes	Sl.No.	Attributes
1	Src_bytes	22	diff_srv_rate
2	Dst_bytes	23	dst_host_srv_diff_host_rate
3	logged_in	24	num_file_creations
4	service	25	same_srv_rate
5	dst_host_srv_count	26	is_guest_login
6	dst_host_same_src_port_rate	27	num_compromised
7	dst_host_serror_rate	28	rerror_rate
8	dst_host_diff_srv_rate	29	dst_host_srv_rerror_rate
9	dst_host_srv_serror_rate	30	srv_diff_host_rate
10	dst_host_count	31	num_root
11	srv_count	32	srv_rerror_rate
12	count	33	wrong_fragment
13	dst_host_same_srv_rate	34	num_failed_logins
14	flag	35	num_access_files
15	serror_rate	36	num_shells
16	duration	37	urgent
17	hot	38	su_attempted
18	protocol_type	39	land
19	srv_serror_rate	40	num_outbound_cmds
20	dst_host_rerror_rate	41	is_host_login
21	root_shell		

SHAP is applied to the Random Forest classifier, as a wrapper method for feature selection, and the most important features were identified and ranked in descending order of importance, as listed in Table 1 [6]. As a baseline, the DNN is trained on all 41 features of the imbalanced NSL-KDD dataset, achieving an overall F1-score of 86% with a training time of 79 seconds. The individual F1-score for the minority attacks R2L and U2R is 86% and 48% respectively. Feature selection was then applied to the model to obtain individual attack outputs and evaluate improvements with the proposed method. The results before and after incrementally adding features from Table 1 to reach the optimum configuration are presented in Table 2.

According to Table 2, implementing the top 26 important features from the SHAP explanation method into the proposed method results in an increase in the F1-score of U2R from 48% to 52%, while the F1-score of Probe decreases from 99% to 98%, and R2L decreases from 86% to 83%.

To enhance performance, the next feature **num_compromised** from Table 1 is added, resulting in an overall F1-score improvement to 90%. Specifically, the U2R F1-score increased from 52% to 70%. Additionally, the R2L F1-score rose to 85%, although it remains lower than the 86% achieved by the model with all 41 features.

Table 2. F1-scores for five-class classification on the imbalanced dataset with different feature sets.

Target classes	41 features	26 features	27 features	Top 27 + num_failed_logins + wrong_fragment + su_attempted
Normal	99	99	99	99
DoS	100	100	100	100
Probe	99	98	98	99
R2L	86	83	85	86
U2R	48	52	70	70
Overall F1-score (%)	**86**	**87**	**90**	**91**
Compilation time (seconds)	**79**	**72**	**72**	**72**

To achieve better results, additional features from Table 1 are incrementally included. However, this did not lead to further improvement in the F1-score. It indicates that the initial 27 features identified by the SHAP explanation feature selection method are crucial for predicting the output. The addition of more features from the SHAP list does not enhance the F1-score; rather, it only adds complexity to the DNN model.

It can be inferred from Table 2 that the training time for the model with 27 features is only 72 seconds, while for 41 features, it is about 79 seconds. This leads to an 8.86% reduction in training time when feature selection is applied.

To further improve the results of R2L and U2R attacks, an attempt is made to incorporate features based on the characteristics of these attacks, instead of selecting from the SHAP list. These features are added to the top 27 features and then applied to the model.

The **su_attempted** attribute in the NSL-KDD dataset indicates whether the "su" command (which is used to switch user accounts) was attempted or not. This attribute is relevant in the context of detecting certain types of attacks, particularly those related to unauthorized access and privilege escalation like the R2L attacks where an attacker tries to gain unauthorized access to a system remotely. This can involve attempting to switch user accounts using the "su" command, among other methods. Therefore, the su_attempted attribute can be important in detecting R2L attacks because it pro-vides information about attempts to switch user accounts, which may indicate an unauthorized user trying to escalate their privileges on the system.

In U2R attacks, an intruder tries to gain superuser or root-level privileges on a system. Multiple failed login attempts could indicate an attacker trying to guess or crack passwords to escalate privileges. For R2L attacks, which involve unau-thorized access attempts from a remote location, the **num_failed_logins** feature is relevant. Brute force attacks, where an attacker systematically tries different passwords, may result in multiple failed login attempts. Since the **num_failed logins** feature in the NSL-KDD dataset represents the number of failed login attempts, it can be relevant in the context of detect-ing both U2R (User-to-Root) and R2L (Remote-to-Local) attacks.

Adding the **su_attempted and num_failed_logins** features is important in detecting R2L attacks and can improve the R2L F1-score without affecting the U2R F1-score, even though using only 27 features did not result in an overall F1-score improvement compared to using all 41 features, which achieved an 86% R2L F1-score.

The F1-score of the Probe attack with 41 features was 99%, surpassing the performance achieved with the previous 27 features. To enhance the results for the Probe attack class, we should incorporate features that are more crucial for its detection, rather than relying solely on the SHAP list. Probe attacks typically involve attempts to gather network information, often through scanning and probing for vulnerabilities, such as port scanning or OS fingerprinting. The **wrong_fragment** feature could be relevant in the context of probe attacks, as unusual patterns in wrong fragments might indicate abnormal behavior. Adding this feature could enhance the model's discriminative power, making it more effective at identifying instances of probe attacks.

After incorporating these three additional features into the top 27 features, an optimal overall F1-score of 91% is achieved, as depicted in Table 2. The F1-score for Probe has increased to 99%, and the F1-score for R2L to 86%, all while maintaining the same compilation time. Thus, a total of 30 features were necessary to attain the optimal outcome, with 27 features selected from the SHAP list and the remaining three features added based on the attributes of the under performing attacks. Further addition of features does not yield any additional improvement in the results.

Across the five-class evaluation, detection of Normal, DoS, and Probe traffic remained consistently strong , while R2L performance held steady in the range of 83–86%. U2R showed notable improvement, rising from 48% to 70% with the optimized reduced feature set. Although these refinements did not alter the overall conclusions, they show that SHAP-driven selection can be complemented by domain knowledge to improve specific attack detections.

Thus Table 2 shows that SHAP-based feature selection reduced the feature set to 26, yielding an F1-score comparable to the baseline while lowering the computational cost. Further refinement by adding a few additional features confirm that SHAP-based selection effectively reduces dimensionality on the imbalanced dataset while maintaining or improving predictive performance.

Despite these improvements, a closer look at the per-attack results important gaps that still remained. The analysis focuses on U2R and R2L minority categories. In the NSL-KDD dataset, U2R includes buffer_overflow, loadmodule, perl, and rootkit, while R2L includes ftp_write, guess_passwd, imap, multihop, phf, spy, warezclient, and warezmaster.

When using the top 26 features, the F1-score of imap has increased from 44% to 50%, and warezmaster from 60% to 75%. For the U2R attack, buffer_overflow, the F1-score is 31% compared to the 37% obtained with 41 features. To enhance the results, features based on the characteristics of underperforming attacks were incrementally added one by one from Table 1, to these 26 features. The buffer_overflow attack involves exploiting software vulnerabilities by overflowing a program's memory buffer, leading to unintended behavior or system compromise. Therefore, the **num_compromised** feature has been added for its efficient detection.

After adding the num_compromised feature(27 features), the F1-score for U2R attack buffer_overflow improved from 31% to 48%, while for R2L attacks like imap and warezclient, it increased to 57% and 87% respectively. The num_compromised feature likely represents the number of compromised systems or devices involved in network activity. U2R and R2L attacks often involve attempts to compromise and gain control over systems. Including this feature allows the model to learn patterns associated with the extent of compromised systems, aiding in the identification of malicious behavior and improving the detection rate of U2R and R2L. However, the results for guess_passwd decreased to 90%, with no change in the results of other attacks.

After adding **su_attempted and num_failed_logins** to the top 27 SHAP-selected features, the model achieved an F1-score of 55% for the buffer_overflow attack, 100% for guess_passwd, and 67% for imap. Further addition of features did not yield significant gains. The best overall outcome was therefore obtained with a total of 29 features, consisting of the top 27 SHAP-ranked attributes along with two additional features chosen for their relevance to minority attacks.

Additionally, the inclusion of features does not significantly enhance F1 scores for certain individual attacks. Among these four levels, the 29-feature selection level yielded relatively positive results for buffer_overflow, guess_passwd, and imap.

Although the individual attack results in Table 3 are improving, there is a slight reduction in the case of warezmaster and warezclient. This may be the reason why the overall F1-score is not improving when considering the same feature in predicting U2R and R2L attacks in the earlier five-class classification.

Table 3. Per-attack F1-scores for U2R and R2L classes across different feature selection stages (baseline: 41 features; reduced: 26–29 features)

Attack classes	41 Features	26 Features	27 Features	27 + 2 Additional Features
U2R Attacks				
buffer_overflow	37%	31%	48%	**55%**
loadmodule	0%	0%	0%	0%
perl	0%	0%	0%	0%
root_kit	0%	0%	0%	0%
R2L Attacks				
ftp_write	0%	0%	0%	0%
guess_passwd	92%	91%	90%	**100%**
imap	44%	50%	57%	**67%**
multihop	0%	0%	0%	0%
phf	0%	0%	0%	0%
spy	0%	0%	0%	0%
warezclient	86%	82%	**87%**	85%
warezmaster	60%	**75%**	**75%**	73%

In all these feature selection levels, the loadmodule, perl, root_kit, ftp_write, phf, and spy still yield zero results. Even with additional features, the F1 scores for these attacks remain at zero. In contrast, attacks such as guess_passwd (100%) and imap (67%) benefited more clearly from SHAP-selected features. This highlights a critical limitation that although SHAP improved the overall F1 score and efficiency, the model was unable to learn from extremely scarce classes in the imbalanced dataset.As a solution, we performed random oversampling to upsample the minority attack class data before applying it to the model.

In this diagnostic experiment, random oversampling was used to expand each of the 22 minority attack classes so that they matched the majority Normal class (67,343 samples). This balancing was applied before performing the 70/30 train–test split, meaning that both the training and test sets contained an equal number of samples across all classes. The purpose of this setup was not to simulate deployment, but to examine how SHAP-based feature selection performs when minority classes are adequately represented, providing insight into its potential under balanced conditions.

While Table 3 shows that SHAP-based feature reduction improves the overall F1-score (23 target classes) and efficiency, the disparity between weighted and macro F1-scores in Table 4 illustrates that severe imbalance still limits minority-class detection. Specifically, the weighted F1-score on the imbalanced dataset is 0.99, but the macro F1-score is only 0.61, confirming that the overall gains are dominated by majority classes such as Normal and DoS. To examine whether SHAP continues to provide benefits when minority classes are adequately represented, oversampling was applied as a diagnostic experiment. Under this condition, macro F1 improved to 0.98 while weighted F1 remained at 0.98, and SHAP retained its effectiveness, as detailed in Table 4. The results from the sampling process are presented in Table 5.

Table 5 shows the results of the model on upsampled datasets with 41 and 27 features. The F1-score for less represented attack classes, such as loadmodule, root_kit, multihop, phf, and spy, improved to 100%. Additionally, the F1-score for ftp_write increased from 0% to 97%, and that of perl improved to 90% when using the top 27 features from SHAP explanation. This suggests that SHAP explanation is effective in identifying crucial features for predicting minority classes compared to other feature selection methods. Furthermore, the limited samples in the training dataset have an impact on DNN model building.

The F1-score comparison between the models utilizing the top 27 features and the complete set of 41 features reveals an intriguing finding. For the majority of attacks, both models yield identical F1-scores; however, certain exceptions, such as perl, warezclient, and warezmaster, underscore a potential for improvement. To address this, an augmentation is introduced by incorporating two additional features, su_attempted and num_failed_logins, into the initial 27 features. The subsequent application of this refined feature set to the Deep Neural Network (DNN) yields promising outcomes. Specifically, the perl attack witnesses a substantial enhancement, achieving an impressive 97% accuracy. Similarly, for the warezmaster attack, the results exhibit notable improvement, ascending from 92% to 97%. Encouraged by these advancements, further endeavors are undertaken to refine the model's predictive capabilities. This

involves a strategic addition of new features based on the unique characteristics of each attack, aiming to maximize the model's effectiveness in distinguishing and classifying diverse cyber threats.

Table 4. Comparison of weighted and macro F1-scores on imbalanced vs. oversampled NSL-KDD datasets

Dataset Setting	Weighted F1	Macro F1
Imbalanced	0.99	0.61
Oversampled	0.98	0.98

Table 5. Per-attack F1-scores for U2R and R2L classes across feature selection stages under oversampled conditions

Attack classes	41 Features	26 Features	27 Features	27 + 2 Additional features
U2R Attacks				
buffer_overflow	99%	99%	99%	**99%**
loadmodule	100%	100%	100%	**100%**
perl	95%	95%	93%	**97%**
root_kit	100%	100%	100%	**100%**
R2L Attacks				
ftp_write	96%	96%	96%	**96%**
guess_passwd	96%	97%	**97%**	94%
imap	93%	96%	**96%**	91%
multihop	100%	100%	100%	**100%**
phf	100%	100%	100%	**100%**
spy	100%	100%	100%	**100%**
warezclient	93%	92%	93%	**94%**
warezmaster	94%	92%	92%	**97%**

Similarly, a notable achievement is observed with the warezclient attack, where the F1 score peaks at an impressive 94%, marking a subtle improvement from the previous 93%. Interestingly, the F1 scores for the remaining attacks in this subset of 30 features exhibit minimal deviation. Despite the overall success, attempts to enhance the F1 scores further through the addition of more features to this set do not yield significant improvements. The model's performance appears to plateau, emphasizing the optimal effectiveness achieved with the selected 30 features. This consolidation of results underscores the importance of feature selection precision and highlights the point of diminishing returns in further feature augmentation.

5.1. Discussion

The results indicate that SHAP-based feature selection can reduce the dimensionality of the NSL-KDD dataset without compromising detection capability. The baseline DNN trained on all 41 features of the imbalanced dataset achieved an overall F1-score of 86%. After applying SHAP, the feature space was narrowed to 26 inputs, yielding nearly identical performance while lowering computational demand. This confirms that a more compact representation is sufficient for modelling attack behaviours.

Minority classes such as U2R and R2L remained challenging under the imbalanced dataset, with several attacks scor-ing close to zero. Introducing oversampling alleviated this issue by creating a more balanced distribution, which in turn raised the per-class F1-scores for rare attacks. Importantly, SHAP continued to play a role even under balanced condi-tions: the reduced feature set not only maintained but, in many cases, improved detection accuracy. These observations suggest that SHAP provides value beyond interpretability—it actively contributes to building leaner and more effective models.

5.2. Limitations

This work has several constraints that shape how the results should be interpreted. The experiments rely solely on the NSL-KDD dataset, which, while widely used, does not capture the full variety of modern threats such as zero-day exploits and advanced persistent attacks. The choice of a shallow DNN architecture was guided by practical concerns of training efficiency during repeated feature selection, rather than being optimized for absolute accuracy. Oversampling was applied before splitting the data into training and testing partitions, which produced balanced test sets. This approach was intended to highlight the influence of SHAP under more equitable conditions, but it does not fully mirror the imbalance present in real network traffic. Lastly, although the study compared SHAP with Random Forest and DNN classifiers, extending the evaluation to additional models such as XGBoost or LightGBM would provide a broader perspective.

6. Conclusion and Future Scope

This work demonstrates that explainability-oriented feature selection can make a measurable difference in intrusion detection. By applying SHAP, the original 41-dimensional NSL-KDD dataset was reduced to a smaller, more focused feature set while preserving overall performance and improving computational efficiency. The compact representation retained strong detection for majority classes and, when augmented with a few targeted features, substantially enhanced recognition of hard-to-detect U2R and R2L attacks. These gains show that SHAP contributes not only to interpretability but also to tangible improvements in detection outcomes.

The study also illustrates how imbalance of data shapes IDS performance. Several minority attacks could not be detected reliably under the natural distribution, but their performance improved markedly once oversampling was introduced. While this balancing step provided diagnostic insight rather than a deployment-ready solution, it highlighted how feature selection and data-level strategies can complement one another.

Looking ahead, validation on recent datasets such as CICIDS2017 or UNSW-NB15 will be crucial for testing robustness against modern attack vectors. Further exploration with more powerful classifiers, including XGBoost and LightGBM, and deeper neural architectures could confirm whether the benefits observed here are model-independent. Finally, coupling SHAP with higher-level interpretability approaches, such as concept-based methods, could allow IDS models to deliver not only better predictions but also human-understandable explanations, making them more practical for real-world cybersecurity defence.

References

[1] B. Mukherjee, L.T. Heberlein, and K.N. Levitt. Network intrusion detection. *IEEE Network*, 8(3):26–41, 1994.

[2] Chih-Fong Tsai, Yu-Feng Hsu, Chia-Ying Lin, and Wei-Yang Lin. Intrusion detection by machine learning: A review. *Expert Systems with Applications*, 36(10):11994–12000, December 2009.

[3] Mohamed Amine Ferrag, Leandros Maglaras, Sotiris Moschoyiannis, and Helge Janicke. Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study. *Journal of Information Security and Applications*, 50:102419, 2020.

[4] Christoph Molnar. *Interpretable Machine Learning.* Lean Pub, Germany, 2 edition, 2022.

[5] Scott M. Lundberg, Gabriel Erion, Hugh Chen, Alex DeGrave, Jordan M. Prutkin, Bala Nair, Ronit Katz, Jonathan Himmelfarb, Nisha Bansal, and Su-In Lee. Explainable AI for Trees: From Local Explanations to Global Under-standing. *arXiv:1905.04610 [cs, stat]*, May 2019. arXiv: 1905.04610.

[6] L Dhanabal and SP Shantharajah. A study on nsl-kdd dataset for intrusion detection system based on classification algorithms. *International journal of advanced research in computer and communication engineering*, 4(6):446–452, 2015.

[7] Rafa Alenezi and Simone A Ludwig. Explainability of cybersecurity threats data using shap. In *2021 IEEE Sympo-sium Series on Computational Intelligence (SSCI)*, pages 01–10. IEEE, 2021.

[8] Elike Hodo, Xavier Bellekens, Andrew Hamilton, Christos Tachtatzis, and Robert Atkinson. Shallow and deep networks intrusion detection system: A taxonomy and survey. *arXiv preprint arXiv:1701.02145*, 2017.

[9] Fadi Thabtah, Suhel Hammoud, Firuz Kamalov, and Amanda Gonsalves. Data imbalance in classification: Experi-mental evaluation. *Information Sciences*, 513:429–441, March 2020.

[10] Bartosz Krawczyk. Learning from imbalanced data: open challenges and future directions. *Progress in Artificial Intelligence*, 5(4):221–232, November 2016.

[11] Srinivas Mukkamala and Andrew H Sung. Feature selection for intrusion detection with neural networks and support vector machines. *Transportation research record*, 1822(1):33–39, 2003.

[12] Bekir Karlik. The positive effects of fuzzy c-means clustering on supervised learning classifiers. *Int. J. Artif. Intell. Expert Syst.(IJAE)*, 7:1–8, 2016.

[13] Scott M. Lundberg, Gabriel Erion, Hugh Chen, Alex DeGrave, Jordan M. Prutkin, Bala Nair, Ronit Katz, Jonathan Himmelfarb, Nisha Bansal, and Su-In Lee. From local explanations to global understanding with explainable AI for trees. *Nature Machine Intelligence*, 2(1):56–67, January 2020. Number: 1 Publisher: Nature Publishing Group.

[14] Raquel Rodr´ıguez-Pe ́rez and Ju ̈rgen Bajorath. Interpretation of machine learning models using shapley values: application to compound potency and multi-target activity predictions. *Journal of computer-aided molecular design*, 34:1013–1026, 2020.

[15] Wilson E. Marc´ılio and Danilo M. Eler. From explanations to feature selection: assessing shap values as feature selection mechanism. In *2020 33rd SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*, pages 340–347, 2020.

[16] Anagha A.S., Ciza Thomas, and N. Balakrishnan. Optimized intrusion predictions through feature selection meth-ods. *Computers & Security*, 157:104541, 2025.

[17] Ravi Vinayakumar, Mamoun Alazab, KP Soman, Prabaharan Poornachandran, Ameer Al-Nemrat, and Sitalakshmi Venkatraman. Deep learning approach for intelligent intrusion detection system. *Ieee Access*, 7:41525–41550, 2019.

[18] Leo Breiman. Random forests. *Machine Learning*, 45:5–32, 2001.

[19] Gilles Louppe. Understanding Random Forests: From Theory to Practice. *arXiv:1407.7502 [stat]*, June 2015. arXiv: 1407.7502.

[20] W. E. Marcilio and D. M. Eler. From explanations to feature selection: assessing shap values as feature selection mechanism. In *2020 33rd SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI)*, pages 340–347, Los Alamitos, CA, USA, nov 2020. IEEE Computer Society.

[21] Guofei Gu, Prahlad Fogla, David Dagon, Wenke Lee, and Boris Skoric´. Measuring intrusion detection capability: An information-theoretic approach. In *Proceedings of the 2006 ACM Symposium on Information, computer and communications security*, pages 90–101, 2006.

[22] Tuan A Tang, Lotfi Mhamdi, Des McLernon, Syed Ali Raza Zaidi, and Mounir Ghogho. Deep learning approach for network intrusion detection in software defined networking. In *2016 International Conference on Wireless Networks and Mobile Communications (WINCOM)*, pages 258–263, 2016.
[23] Ciza Thomas. Improving intrusion detection for imbalanced network traffic: Improving intrusion detection. *Security and Communication Networks*, 6(3):309–324, March 2013.
[24] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*, 16:321–357, June 2002.
[25] Yadigar Imamverdiyev and Fargana Abdullayeva. Deep learning in cybersecurity: Challenges and approaches. *International Journal of Cyber Warfare and Terrorism*, 10:82–105, 03 2020.

Authors' Profiles

Anagha A. S. received her B.Tech in Electronics and Communication Engi-neering from LBS Institute of Technology for Women, Thiruvananthapuram, in 2008, and her M.E. in Commu-nication Systems from Anna University, Chennai, in 2013. She is currently pursuing her Ph.D. in cyber security at the College of Engineering, Trivandrum, India. She has over nine years of experience as an Assistant Professor in Electronics and Communication Engineering. Her research focuses on Explainable AI, machine learning, and intrusion detection, with particular emphasis on trusted models for minority attack detection. She has published in International journals and conferences and contributed a book chapter in cybersecurity.

Ciza Thomas, Prof. Ciza Thomas is presently serving as the Vice Chancellor of Digital University Kerala. She has earlier held the positions of Vice Chancellor of APJ Abdul Kalam Tech-nological University and Kerala University. She earned her B.Tech. and M.Tech. degrees from the College of Engineering, Trivandrum, and her Ph.D. from the Indian Institute of Science (IISc), Bangalore. Her academic and research expertise spans Cybersecurity, Artificial Intelligence, Image Processing, Pattern Recognition, and Data Mining.

Prof. Thomas has had a distinguished career in the Government Engineering Colleges of Kerala, serving in various capacities as Assistant Professor, Associate Professor, Professor, and Principal. She has also contributed significantly to academic administration as Senior Joint Director and later as Director (in charge) of the Di-rectorate of Technical Education, Kerala. Her contributions have been recognised through prestigious honours, including the Kerala State e-Governance Award (2014) and the Appreciation Award of the Higher Education Department, Government of Kerala (2010).

Sreelatha G. is a Professor in the Department of Electronics and Communica-tion Engineering at the College of Engineering, Trivandrum, India. She received her B.Tech degree in Applied Electronics and Instrumentation Engineering from the College of Engineering, Trivandrum, in 1996, the M.Tech degree in Electronic Design and Technology from the Indian Institute of Science, Bangalore, in 2006, and the Ph.D. degree in Electronics and Communication Engineering from the National Institute of Technology, Calicut, Kerala, in 2017.

She has extensive teaching and research experience, having guided numerous postgraduate and doctoral stu-dents. Her major research interests include signal processing, image and video processing, medical image pro-cessing, VLSI design, and machine learning. She has published widely in international journals and conferences and contributed to several funded research projects in these areas.

How to cite this paper: Anagha A. S., Ciza Thomas, Sreelatha G., "Enhancing Intrusion Detection for Minority Attack Classes: A SHAP-Based Feature Selection Approach with Deep Neural Networks", International Journal of Wireless and Microwave Technologies(IJWMT), Vol.16, No.1, pp. 50-62, 2026. DOI:10.5815/ijwmt.2026.01.04