

ANALYZING ENCRYPTED TRAFFIC ANALYSIS USING EXPLAINABLE AI

SHUDHAMATI MALI¹, SHREYA AGARWAL¹, RUCHA CHINTAWAR¹, ASWANI
KUMAR CHERUKURI^{1*}

ABSTRACT. The increasing reliance on machine learning (ML) for network traffic classification highlights a critical need for not only predictive accuracy but also transparent, interpretable models that foster trust and accountability in cybersecurity applications. This study presents a reproducible framework that applies tree-based and kernel-based classifiers—Random Forest, Decision Tree, and Support Vector Machine—to widely used VPN, phishing, and botnet traffic datasets. By systematically integrating both SHAP (SHapley Additive exPlanations) and LIME (Local Interpretable Model-Agnostic Explanations), we provide detailed, actionable insights into model decision processes and dominant feature contributions. Unlike prior works, our approach consolidates and validates established domain knowledge while mapping global and local model explanations to specific network protocol behaviors and attack vectors. Furthermore, we discuss the practical implications of our findings for feature engineering and lightweight model design. While the use of legacy datasets offers continuity with existing benchmarks, we directly address their limitations and outline paths for future work using contemporary data. This work serves as a transparent, practitioner-oriented interpretability guide for deploying ML-based traffic classifiers in evolving security environments.

Keywords. Explainable AI, Machine Learning, LIME, Network traffic Classification, SHAP

2020 Mathematics Subject Classification. Primary 46L55; Secondary 44B20

1. INTRODUCTION

Securing digital environments in today's interconnected world demands precise and reliable network traffic classification. Effective cybersecurity not only requires the detection of known threats and intrusions, but also the identification of emerging malicious patterns hidden within complex network flows. Machine learning (ML) models have become invaluable for this purpose, as they are capable of categorizing diverse traffic types—including benign, malicious, encrypted, and obfuscated flows—with high predictive accuracy. However, as these models grow in complexity, their inherent “black box” nature raises new challenges: lack

Date: Received: Sep 14, 2025; Revised: Oct 14, 2025; Accepted: Jan 10, 2025.

* Corresponding author

© The Author(s) 2025. This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License. To view a copy of the licence, visit <https://creativecommons.org/licenses/by-nc-nd/4.0/>.

of transparency and interpretability frequently undermines both trust in their decisions and the ability to debug or refine them for practical deployment.

This transparency gap is a significant barrier for real-world cybersecurity applications. Stakeholders must be able to understand not just what the model predicts, but why a particular classification was reached—especially in critical contexts such as intrusion detection, VPN traffic analysis, and phishing protection. Explainable AI (XAI) techniques, such as SHAP (SHapley Additive exPlanations) and LIME (Local Interpretable Model-Agnostic Explanations), have recently emerged as practical solutions for demystifying ML decisions, by revealing the influence of individual features on model outputs both globally and for specific predictions.

Despite the growing adoption of XAI in various domains, there remains a need for systematic, practitioner-focused studies that not only apply these methods to standard cybersecurity datasets, but also map the resulting explanations back to real-world network behaviors and attack strategies. Most prior work has demonstrated the usefulness of XAI for interpreting ML models in traffic classification; however, it is equally important to consolidate, validate, and contextualize these insights to serve both researchers and network security professionals.

In this study, we present a reproducible, step-by-step framework integrating SHAP and LIME explanations with Random Forest, Decision Tree, and Support Vector Machine classifiers for the analysis of VPN, phishing, and network intrusion datasets. Rather than introducing novel algorithms, our primary aim is to enhance the interpretability and practical understanding of existing models in representative security contexts. By connecting feature importance profiles and explanation visualizations to known network behaviors, our work offers concrete guidance for model optimization, feature engineering, and transparent cybersecurity practices.

This approach not only provides an accessible interpretability pipeline but also directly supports transparent, fair, and actionable machine learning deployments in cybersecurity. Limitations arising from the use of legacy datasets and established ML methods are acknowledged, with suggestions for future work oriented towards emerging datasets and adversarial robustness. Ultimately, this study aims to reduce the “black box” barrier and empower practitioners to build, trust, and improve ML-driven network defense systems.

The main novelty of this study is the development of a transparent, reproducible workflow that systematically applies explainable AI techniques—specifically SHAP and LIME—to standard machine learning classifiers for network traffic analysis. Unlike previous works that focus solely on performance metrics or present explanations without domain context, we explicitly map XAI findings to concrete network protocol behaviors and known attack vectors, offering actionable interpretations for practitioners. While the underlying ML models and datasets are well-established, our contribution is the accessible integration, comprehensive explanation, and practical guidance for deploying interpretable models in real-world cybersecurity scenarios. This work sets a foundation for further research on model transparency, feature selection, and adversarial robustness in traffic classification.

Rest of the paper is organized as follows: Section 2 provide literature analysis. Section 3 provide details of ML and explainability techniques. Section 4 provide architecture of the proposed work. Section 5 provides results followed by conclusions and references.

2. LITERATURE

The increased adoption of encryption protocols such as TLS, HTTPS, and VPN has complicated the task of securing a network because most traditional methods of traffic analysis cannot process encrypted data properly [1]. Under such circumstances, the ML-based models have presented hope, but at the same time, concern has arisen due to the black box nature of such models relating to trust and interpretability. This gap is increasingly being bridged by the use of explainable AI (XAI) to acquire insight into how the machine learning models determine their respective decisions that deploy in an analysis of the encrypted traffic. Recent advancements in encrypted traffic classification have focused on leveraging deep learning techniques, such as Convolutional Neural Networks (CNNs), to automatically extract features from raw packet data. Despite their effectiveness, CNNs still face challenges related to high training times and limited temporal feature extraction. In response, Zhou et al. propose a novel model combining DenseNet and Bidirectional Gated Recurrent Units (BiGRU) [2]. DenseNet enhances spatial feature extraction by improving information flow between layers and reducing parameters. BiGRU addresses the limitations of unidirectional models by capturing both forward and backward dependencies, thus improving temporal feature extraction. This model demonstrates superior performance in classifying encrypted traffic, showing significant improvements in accuracy, precision, and training time over existing methods.

The proposed methodology, known as axiomatic attribution, is designed to provide interpretable explanations of how input features contribute to the output of deep networks [3]. The authors emphasize the significance of satisfying specific axioms to ensure that the attribution method is consistent and reliable, making it applicable to a wide range of machine learning models. The approach is characterized by the preservation of fairness and transparency, even in the complexity of deep neural networks. Through the formalization of the concept of attribution, it contributes to better interpretability for black-box models and offers insight into fields such as healthcare, finance, and autonomous systems that require model explainability.

Phishing attacks are one of the most critical cyber threats, owing to the dependency on online transactions and communication systems. Bhupathi et al. [4] conducted a study where they used some machine learning algorithms, such as RF, DT, LR, and SVM, in detecting phishing URLs. In that regard, the RF model resulted in the highest accuracy of 98.55%, indicating the robustness of the model when it comes to the task of phishing detection. To improve the interpretability of the model's predictions, XAI methods, namely SHAP and LIME, were used. SHAP helps explain the global contribution of features, while LIME provides explanations for localized aspects, thereby boosting user trust and transparency

in the detection process. The study also focused on more sophisticated phishing techniques like domain spoofing and path manipulation, and that is why a combination of machine learning with XAI would make the system improve accuracy and trustworthiness for phishing URL detection systems. This work provides an essential foundation for techniques like SHAP and LIME, which aim to make machine learning models more transparent and understandable for practitioners and researchers. The paper by Do et al. [5] identifies network security and mitigation towards possible breaches as some of the primary uses of anomaly detection in the Internet of Things (IoT) space. Generally, traditional models such as machine learning and deep learning models have been reported to achieve higher accuracy in anomaly detection but tend to receive criticism as black-box models due to unavailability of explanations. The authors, therefore, consider the feasibility of xAI algorithms for IoT anomaly detection as a way to get around this dilemma. Their work seeks to explain four XAI algorithms as applied to supervised and unsupervised anomaly detection models and uses two datasets: MQTTset and CIC IDS 2017. The key results show that the best and most trustworthy XAI algorithm for explaining the outcome of anomaly detection is SHAP while the image-based XAI algorithms are not considered sketchy for network traffic. This paper serves the purpose of tackling the problems posed by accurate and intelligent anomaly detection models in terms of being able to be understood and comprehended for real-life uses.

Recent studies of advanced techniques involve malware detection via memory analysis together with machine learning (ML), deep learning (DL), and XAI to obtain higher accuracy and understandability. A number of methods prove effective in converting memory images to grayscale and with the use of classifiers such as MLP and SVM, achieving significant accuracies by Ch et al., 2024 [6]. Hybrid models including architectures such as LSTM and GRU have also shown better performance in multi-class malware detection, which indicates that the combination of different neural networks is useful. In addition, XAI techniques like LIME offer interpretability to provide insights into which features are impacting the detection decisions. However, there are challenges, such as the high computational cost of real-time detection and the need for scalable models that can generalize to new malware. These findings underscore the potential of integrating memory analysis, ML, and XAI to improve malware detection and offer insights for enhancing encrypted traffic analysis using similar methodologies.

Another important line of research concerns the security and reliability of machine learning models themselves. Jonnalagadda et al. (2024) [12] investigate data poisoning attacks against convolutional neural networks, demonstrating how small but strategically designed perturbations in the training data can substantially degrade model performance and compromise prediction integrity. Their work highlights the vulnerability of deep learning pipelines to adversarial manipulation and underscores the necessity of integrating explainability and robust training mechanisms in cybersecurity-oriented AI systems.

Zilli et al. [7] have introduced an innovative approach to enhancing the Internet traffic matrix estimation task, which plays a crucial role in several network management tasks such as routing, anomaly detection, and troubleshooting. Since

obtaining accurate data regarding traffic matrix remains a tough job due to networks often encountering incomplete or noisy information about their flow, such problems are mentioned. The authors address this issue by incorporating XAI techniques, such as LIME and Saliency Maps, into the model to provide interpretability and better understanding of model behavior. These methods help identify the most relevant features for the completion process, which significantly reduces the amount of data needed for accurate predictions while maintaining or even improving performance. Applying their proposed methodology to CNN, reconstruction error is decreased as much as by 80% using 75% fewer data. The paper conducts trace-driven studies over real-world datasets and compares their XAI-driven approach with eight cutting-edge TM estimation techniques, showcasing greater results than the other techniques. This research further brings to the fore the power of combining AI and XAI to conduct efficient and transparent network traffic analysis.

Multimodal deep learning architecture known as MIMETIC-ALL was used for the co-classification of both app activity and user activity in network traffic [8]. By applying SHAP and other XAI techniques, the authors interpret the decisions of the model and identify the most influential features contributing to the classification, thereby improving the overall transparency and trustworthiness of the model. This opens the way to more interpretable and reliable traffic classification systems, which are important for the effective management and security of modern networks.

3. MATERIAL AND METHODS

Using an ensemble of a wide variety of methodologies based on Machine Learning (ML) and Explainable Artificial Intelligence (XAI), we have provided high accuracy in the classification of encrypted network traffic together with interpretability of models for their better understanding and acceptance by the network administrator [9, 10, 11].

3.1. Machine Learning Based Models. SVM is a powerful algorithm for supervised learning, that can handle both binary as well as multi-class classification tasks. The data has been chosen for this task of network traffic classification since it is quite effective for handling high-dimensional data. The algorithm provided strong class separation and thus an easily defined decision boundary. SVM depends upon kernel functions that map input data to a higher dimension, hence making it possible to transform non-linearly separable data into linearly separable data. In this study several kernel functions are applied to study different aspects of the network traffic data. The linear kernel was applied in cases where the traffic patterns were linearly separable in the feature space, making it possible to classify in a straightforward manner. Whereas in traffic data there are complex non-linear interaction, the use of an RBF kernel was involved. RBF is applied when dealing with encrypted data because they measure the similarity between the instances of high dimensions. This may turn out important for identifying hidden patterns that one may not even notice from the lower-dimensional space. The polynomial kernel introduced more complexity in the classification

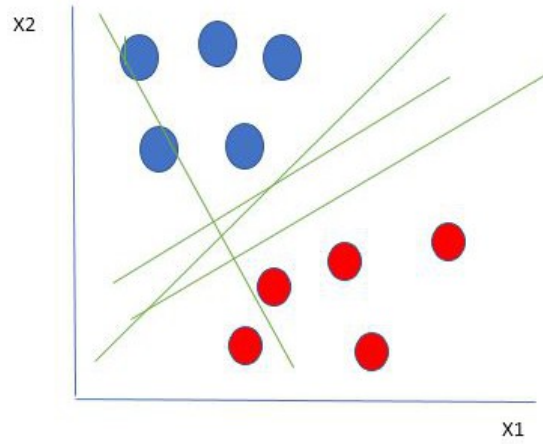


FIGURE 1. Support Vector Machine

problem by bringing in the ability to account for interaction among multiple features, thereby providing better capability for classification in cases where feature interaction was a significant factor. The key goal behind the SVM model was to ensure that the margin between different classes of traffic was maximized, for instance, between VPN and non-VPN as shown in Figure 1. This ensured robust classification, even with tiny variations in the data. Quadratic programming was implemented to solve the optimization problem inherent in SVM. The best decision boundary was obtained by minimizing the hinge loss function, and instances of misclassifications were penalized to make the model perform better on the whole.

Random Forest is an ensemble learning technique whereby it constructs a number of decision trees at training time, and the overall prediction is made by aggregating all the predictions as shown in Figure 2. This ensures that the network traffic classifier learns robustly as well as generalizes properly, a requirement for dealing with complex interactions in features. Bootstrapping and feature randomization allow diversity to be included in the training procedure through this approach. For that reason, the training on each decision tree takes a random subset of data features, helping the model prevent overfitting. Each tree's capturing noise from the data used to train will actually produce somewhat different predictions in the presence of slightly varying subsets they were trained with. One of the advantages of the RF approach is that it allows a direct measure of feature importance. It is possible to determine which features have the most influence on the classification of encrypted network traffic using this feature importance metric. In traffic analysis, some features such as packet size or inter-arrival time may be more important in distinguishing between VPN and non-VPN traffic, and the RF helps identify such factors.

Due to the high interpretability and simplicity, decision trees are widely used. The algorithm splits the dataset into smaller subsets according to the inputs of the features. It chooses the splitting feature at a node that gives the best separation of classes at each step based on metrics used such as Gini impurity

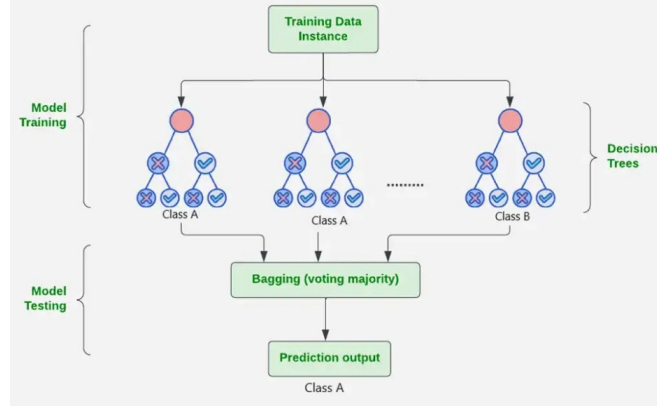


FIGURE 2. Random Forest

or Information Gain. Pruning techniques are used to prevent overfitting. It reduces the complexity of the tree and removes branches that contribute less to the model’s performance on the validation set. Therefore, pruning ensures that the decision tree remains generalizable to unseen data while maintaining its interpretability. One of the main advantages of decision trees is that they are interpretable. Every path from root to leaf represents a decision rule, which can be easily understood by network administrators. Interpretability is very much helpful in cybersecurity, where one must not only know what was the outcome but also the rationale behind the process of taking that decision.

3.2. Explainable AI based Models. Since machine learning models, particularly ensemble models such as RF and complex models such as SVM, are typically ”black-box” in nature, we included XAI methods to make the models more interpretable. The following XAI methods were employed to understand how the models ended up with their predictions.

SHAP values are derived from cooperative game theory, explaining how each feature contributes to the final prediction. SHAP assigns importance scores to features based on their marginal contribution to the final prediction by considering all possible feature combinations. This makes SHAP particularly powerful in understanding how individual features influence model outputs both globally (across the entire dataset) and locally (for specific instances). SHAP values are widely used because they provide a fair and mathematically justified method of attributing importance to each feature. The study utilized two types of SHAP visualizations: the SHAP summary plot (bee swarm plot) and the SHAP bar plot. The SHAP summary plot represents individual feature contributions for each data instance, highlighting variations in how different features impact predictions. This plot helps in identifying whether a particular feature has a consistent impact across different samples or if its influence varies significantly.

Although SHAP offers a global and local feature importance analysis, LIME was also used to improve interpretability, especially at the local level. In contrast to SHAP, which offers a full distribution of feature contributions, LIME aims to explain individual predictions by approximating the complex model with a

simpler, more interpretable one. LIME produces synthetic perturbed data surrounding the given instance and trains a lean surrogate model that imitates how the original model behaves in this localized space. This makes understanding why a certain instance was labeled in a given manner more tangible. For instance, in phishing URL classification, LIME would indicate which of the features—like domain length, number of tokens, or special character appearance—were most responsible for classifying a URL as a phishing URL.

4. ARCHITECTURE AND EXPERIMENTAL SETUP

The architecture outlines the workflow for developing and explaining a network traffic classification mode as shown in Figure 3. First and foremost, information of network traffic is collected from various sources such as IDS, VPN, URL, and probably some more. This raw data is then transformed by a pre processor to handle missing values and discrepancies, and then it is normalized to keep it on the same scale and hence improve the model’s efficiency. Further, the normalized data is then split into the training and testing datasets. The training set is used to create many different machine learning models, like support vector machine, decision tree, and random forest, where each of them uses its capabilities and suffers from complexities. Next, the models will be run on a test to check for their classification accuracy. XAI methods are introduced to address the “black-box” nature of the models. It is inferred that SHAP and LIME are applicable where these methods are used to highlight the ones responsible for the model predictions. In the end, the outcome is a built-in machine learning model alongside its respective XAI explanation that helps to do the correct classification and have a better understanding of the model’s decision-making process. This model emphasizes the basic role of data preprocessing and model options in the effective network traffic classification; at the same time, introducing XAI methods creates more clarity and reliability, therefore, obtaining stronger and more understandable network security solutions.

This research adopts a fused network traffic dataset, using three independent datasets Intrusion Detection System Dataset or ISCX IDS 2012 and ISCX VPN-non VPN 2016 and then the URL dataset or ISCX-URL 2016. Datasets so used are popular in different cybersecurity research domains and constitute many instances of various network traffics, and thus such datasets will really help power machine learning model development by analyzing traffic to their efficiency in carrying out intrusion detection classifications of URL.

The ISCX VPN-nonVPN 2016 dataset is one of the most commonly used for VPN and non-VPN traffic analysis. The ISCX-VPN2016 dataset focuses on distinguishing Virtual Private Network (VPN) traffic from non-VPN traffic, thus addressing the challenges of encrypted communications. This dataset is captured under real-world conditions and contains traffic attributes such as packet size, duration, protocol, and flow statistics as shown in Table 1. It is particularly valuable for evaluating traffic classification models and understanding encrypted traffic patterns, which are critical for modern network security. It greatly supports research focused on securing networks by identifying VPN-related threats.

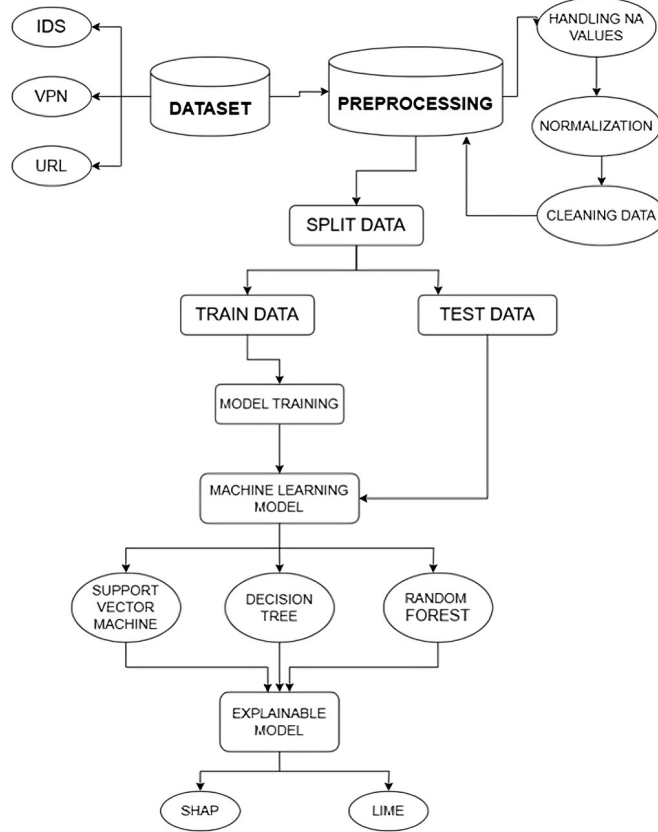


FIGURE 3. Proposed Architecture

TABLE 1. Features and descriptions of the VPN traffic dataset.

Feature	Description
flowPktsPerSecond	Number of flow packets per second.
min_fiat	Minimum forward inter-arrival time.
mean_idle	Mean idle time between flows.
max_biat	Maximum backward inter-arrival time.
class1	Label indicating VPN or non-VPN traffic.

ISCX-URL2016 is specifically designed for URL analysis. It is widely used in the detection of phishing attempts and malicious websites. The dataset categorizes URLs into legitimate, phishing, and malicious with features such as URL length, special character count, and domain-specific patterns. This dataset supports the development of robust models for URL classification, which will effectively prevent phishing attacks and secure web-based interactions. This dataset provides critical features like URL length, top-level domains, and character patterns, which are key indicators in classifying a URL as phishing or benign as shown in Table 2 . It supports efforts to bolster cybersecurity defenses against phishing.

TABLE 2. Features and descriptions of the URL traffic dataset.

Feature	Description
domain_token_count	Number of tokens in the domain part of the URL.
tld	Top-level domain (e.g., .com, .org, .net).
Entropy_URL	Entropy of the entire URL string.
SymbolCount.FileName	Number of symbols in the file name.
URL_Type_obf_Type	Type of URL obfuscation.

The ISCX IDS 2012 dataset is network intrusion detection, capturing all varieties of network traffic activities both in benign and malicious categories. The dataset contains normal user activities, port scans, and denial-of-service attacks. Each instance is labeled with a corresponding class, indicating whether the traffic is normal or attack-type. The data was large initially and contained categorical and numerical values, which were later pre-processed for improved efficiency in the models. It contains attributes such as duration, protocol_type, service and etc as shown in Table 3.

TABLE 3. Features and descriptions of ISCX Botnet dataset.

Feature	Description
duration	Total time length of the connection.
protocol_type	Type of protocol used in communication (e.g., TCP, UDP, ICMP).
service	Network service on the destination (e.g., http, telnet, ftp).
src_bytes	Number of data bytes sent from the source to destination.
flag	Status of the connection (e.g., SF for successful connection).
dst_bytes	Number of data bytes sent from the destination to source.
land	Whether the connection is from/to the same host/port (0 = false, 1 = true).

The combined datasets form a strong basis for this research, since they cover a wide variety of network traffic scenarios. The models developed based on such datasets help address real-world cybersecurity issues, from encrypted traffic analysis to phishing detection and botnet activity monitoring. Leverage these datasets to develop a study that contributes to advancing the science of effective and interpretable solutions in the field of network security.

ISCX IDS 2012, ISCX-VPN2016, and ISCX-URL2016 are standard benchmarks frequently used in prior research, allowing direct comparison with the literature and facilitating reproducibility for both methodological and interpretability studies. These datasets offer detailed, well-vetted labeling and feature sets that are not always available in more recent public datasets, ensuring the reliability of machine learning experiments and feature attribution analyses. Their broad adoption in the literature makes them valuable for benchmarking and reproducibility of interpretability workflows.

In order to ensure high quality and reliability of the datasets during subsequent machine learning analysis, a thorough cleaning process of the three selected

datasets is carried out: VPN and Non-VPN traffic Dataset (ISCX-VPN2016), URL Dataset (ISCX-URL2016), and ISCX Botnet Dataset (ISCX-Bot-2012). Every dataset carries out the following detailed pre-processing steps for dealing with missing values, inconsistencies, as well as data standardization. Missing data entries affect the model’s accuracy significantly as well as its ability to generalize. The existence of null values was determined using `isnull()` in Python to report a summary of all null values for each feature. Depending on the character of the dataset and kind of feature, missing values were imputed using the mean of the respective columns. For example, features such as `packet_size` and `flow_duration` in the ISCX-VPN2016 dataset were handled using mean imputation to keep the numerical ranges consistent.

The three datasets presented features with different scales and units, such as `time_duration` measured in seconds and `packet_count` as discrete counts. Standardization of the range of the said features was done by Min-Max scaling. In order to avoid bias in training models due to dominant larger-scale features, all numerical features were scaled to a 0 to 1 range. Categorical features were encoded to numerical values for compatibility with machine learning algorithms.

Visualization was a critical step in the understanding of structures of the datasets to help guide subsequent feature engineering and model development. For each dataset, detailed exploratory data analysis (EDA) was done using Python visualization libraries like `matplotlib` and `seaborn`. Correlation coefficient function is used for determining those redundant features to eliminate such unwanted redundant information from the stream of variables considered while implementing model training processes in ISCX-VPN2016, ISCX-URL2016, as well as in the dataset of ISCX-Bot-2012. By using correlation coefficients for feature reduction, all three datasets were reduced significantly in dimensionality. The computation became more efficient, the problem of multicollinearity was minimized, and the risk of overfitting during training of models was reduced. Specifically, Pearson correlation coefficients were computed for all feature pairs, and features with absolute correlation values greater than $|r| > 0.9$ were removed to avoid multicollinearity. When two features were highly correlated, the one with a lower correlation to the target class was dropped. After this filtering step, the retained feature counts were 28 for ISCX-VPN2016, 34 for ISCX-URL2016, and 30 for ISCX-IDS2012. This choice of threshold, consistent with standard practice in feature selection, ensured removal of redundant attributes without loss of predictive fidelity.

This would lead to a good foundation in the development of accurate and interpretable machine learning models. These relationships were shown in the visualizations like heatmaps from the correlation analysis and supported the decisions made in this step shown in Figure 4 is of VPN dataset. Three models—RF, DT, SVM—are applied on the classification of traffic patterns for their effectiveness on different datasets.

These models have been selected to be diverse in learning approaches and balance between interpretability and accuracy. All models have been trained and tested with 80:20 split train-test split, and their performance was evaluated based on metrics like accuracy, precision, recall, and F1-score. With the construction

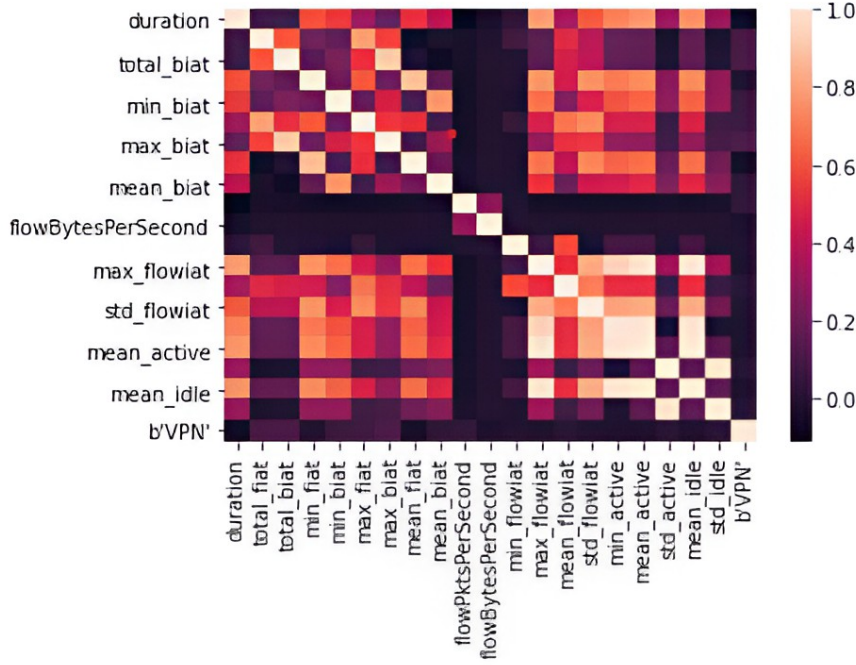


FIGURE 4. HeatMap of Correlation Coefficient

and validation of the machine learning models—Random Forest, Decision Tree, and Support Vector Machine (SVM)—used to classify network traffic patterns, the subsequent critical step is the incorporation of Explainable Artificial Intelligence (XAI) techniques. This is helpful in demystifying the models’ decision-making process, turning the traditional “black-box” nature of these classifiers into an understandable and transparent system. XAI not only builds trust in the deployed model but also enables troubleshooting and tuning of the underlying algorithms, so that the network traffic classification system is effective and answerable.

To accomplish this, two of the most notable XAI methodologies are utilized: SHAP and LIME. These methods are implemented once the machine learning models have been trained and tested, providing a valuable interpretability layer to the system. SHAP, founded on cooperative game theory principles, determines the SHAP value for every feature. This value is the mean contribution of a feature to the model’s prediction over all possible feature input combinations. More simply, the SHAP value measures how much each feature forces the prediction toward a specific class label, increasing or decreasing its probability.

Along with SHAP, this work includes LIME to offer complementary, instance-specific interpretability. Whereas SHAP provides both global and local explanations in terms of Shapley values from cooperative game theory, LIME presents a more natural, localized framework. It functions by creating perturbed copies of a given input instance and watching how the predictions of the model shift. With

this information, LIME approximates a simple, interpretable model to mimic the behavior of the intricate model in the local neighborhood of that instance. This enables us to determine which features were most responsible for a given prediction. By using both SHAP and LIME, the research seeks to marry the strengths of global interpretability with rich, per-instance explanations, giving a more complete picture of the model’s behavior.

5. RESULTS

To evaluate the effectiveness of our classification methods, we make use of four commonly adopted metrics. These are Precision (Pr) or Positive Predictive Value, Recall (Rc) or Sensitivity, Accuracy, and F1 score. Precision is the ratio of correctly identified positive instances in the total instances classified as positive, both correctly and wrongly. Recall calculates how well the model is able to identify all the true positive cases. Accuracy describes the overall accuracy of the model by calculating how many predictions were correct out of all the predictions made. The F1 score offers a balance between precision and recall, particularly helpful if the data is imbalanced, by incorporating both into a single measure of performance. In VPN traffic dataset RF model have highest accuracy of 90 percent compared to other models as shown in Table 4.

TABLE 4. Performance metrics (%) of ML models on the VPN traffic dataset.

ML Models	Accuracy	Precision	Recall	F1-Score
Random forest	90	91	88	89
Decision tree	86	90	85	85
Support vector machine	60	83	59	69

According to the observed performance measures in Table 4, the Random Forest classifier is the top-performing model for the VPN traffic data. For further explanation and justification of its predictions, XAI methods like SHAP and LIME are utilized. Through these techniques, it is possible to discover how specific features affect the model’s choices, providing transparency to the classification procedure. The summary plot is one of the major visualizations and gives a global view of the importance of the features. The SHAP summary plot shown in Figure 5 demonstrates the average impact of each feature on the size of the model output for VPN traffic classification. The x-axis shows the mean SHAP value, which measures the contribution of each feature to the predictions of the model. Greater values show more influence on the model’s output. Out of all the features, total_biat contributes the most, and it means that packet timing in the reverse direction is most important in identifying VPN from non-VPN traffic. Others of notable importance are min_flowiat and std_flowiat, which show that variation and pattern in timing in flows are good predictors. Other important features, such as flowBytesPerSecond and flowPktsPerSecond, also play significantly, and the high values of throughput and packet rate underscore the significance of these

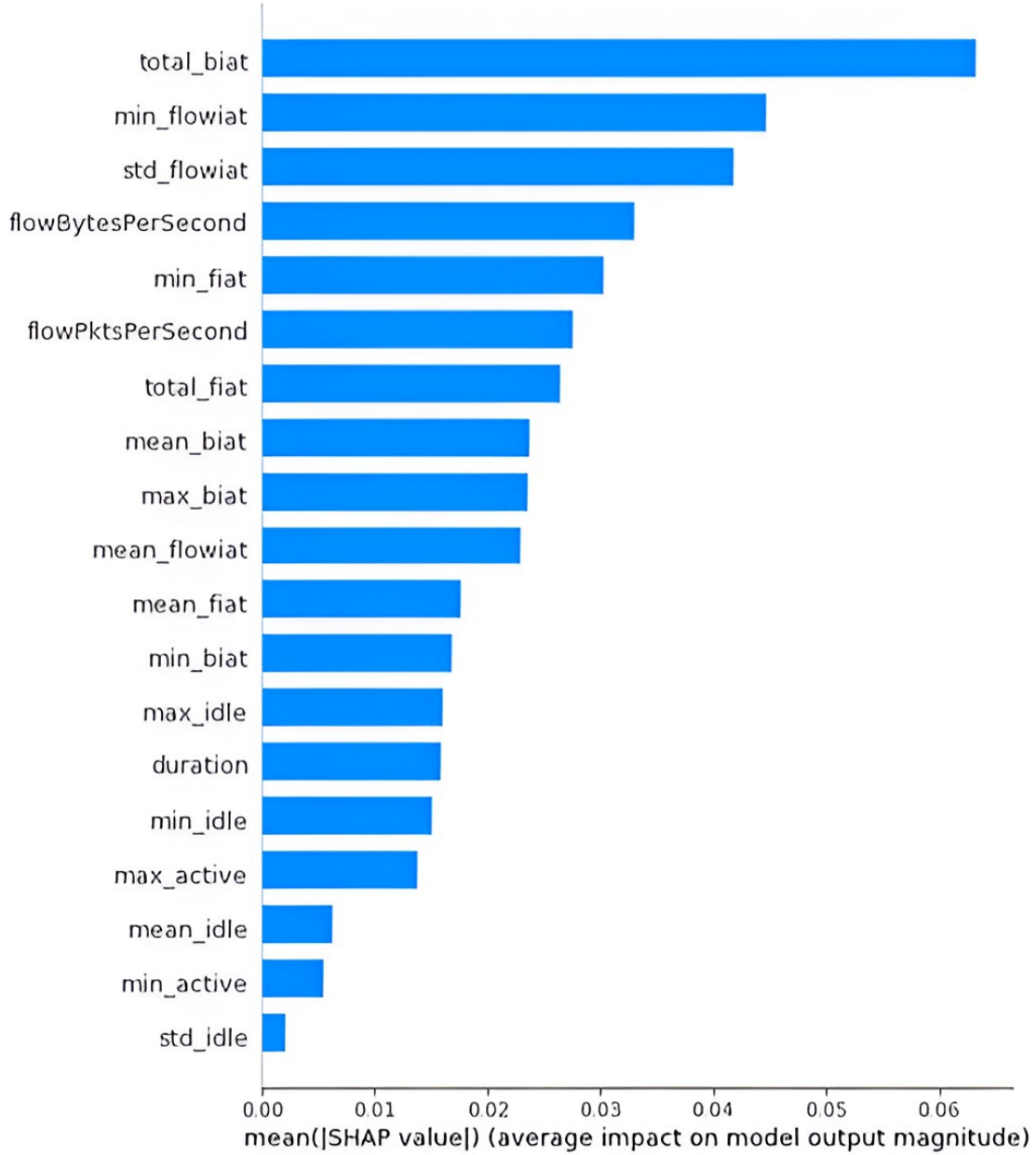


FIGURE 5. Summary Plot of VPN-NON VPN dataset

features. Less significant features, such as `std_idle` and `min_active`, have very low SHAP values, indicating a relatively lesser contribution to the decision. Interpretable analysis assists with understanding the model's behavior and ensuring domain knowledge correspondence between the predictions and network traffic analysis.

Complementing the global insights from SHAP, the LIME plot offers a localized explanation for an individual instance classified by the Random Forest model. The plot in Figure 6 indicates that features such as `flowBytesPerSecond`,

std_flowiat, and flowPktsPerSecond, when close to zero, positively influenced the prediction, suggesting the instance exhibited low data transfer rate and minimal inter-arrival variation—patterns typically associated with VPN traffic. Additionally, timing features like mean_biat and min_biat also played a role in reinforcing the model’s prediction.

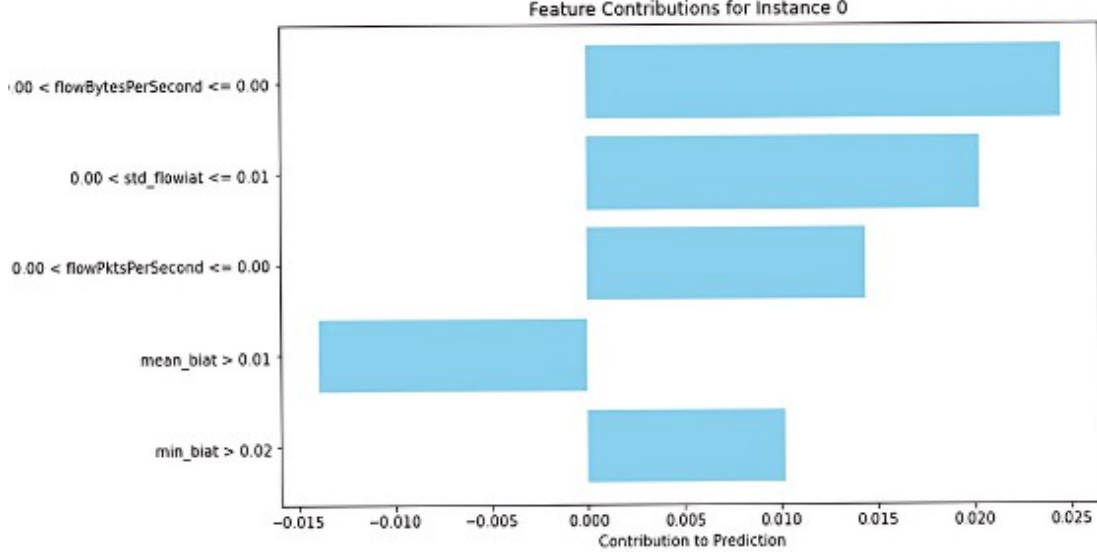


FIGURE 6. Horizontal bar plot of VPN traffic

For the URL dataset, Random Forest model was found to be the top-performing classifier when evaluated using evaluation metrics shown in Table 5. Its behavior was explained using both SHAP and LIME methods. SHAP beeswarm plot gives a worldwide view of feature importance in all predictions shown in Figure 7. It indicates that features like Extension_LetterCount, NumberRate_FileName, NumberRate_Extension, and avgpathtokenlen have the most influence on model output. These features are extracting structural aspects of URLs—such as character length, numeric patterns, and tokenization—that are key in separating benign from malicious URLs. The color scale shows whether high (red) or low (blue) feature values drive predictions towards the positive or negative class, respectively. Concurrently, the LIME horizontal bar plot for instance 0 presents the local contributions of each feature in Figure 8. Features tld, domain_token_count, and SymbolCount_Domain with low values had a negative contribution to the prediction (in the direction of benign), while features executable, host_DigitCount, and longdomaintokenlen had lesser or slight positive contribution.

TABLE 5. Performance metrics (%) of ML models on the URL dataset.

ML Models	Accuracy	Precision	Recall	F1-Score
Random forest	97	97	95	96
Decision tree	94	96	97	97
Support vector machine	88	90	80	85

Some of the features—such as tld, domainlength, SymbolCount_Domain, and longdomaintokenlen—occur in both SHAP and LIME plots, showing their ubiquitous importance globally and locally. This verifies the model’s reliability and its reliance on meaningful URL features in making classification judgments.

For IDS dataset, Decision Tree and Random Forest classifiers have obtained the same and high accuracy. Interpretation hereafter is based on the predictions made by the Random Forest model.

TABLE 6. Performance metrics (%) of ML models on the IDS dataset.

ML Models	Accuracy	Precision	Recall	F1-Score
Random forest	99	99	100	100
Decision tree	99	100	99	100
Support vector machine	98	98	99	98

SHAP force plot shows how each feature impacts the model’s prediction for a particular instance in Figure 9. Attributes like service = http, protocol_type = tcp, and logged_in = False had significant positive impacts, driving the prediction towards classifying the instance as an attack. Features like dst_bytes = 0 (no outgoing data) and service had a minor negative effect, lowering the predicted attack probability. The storyline illustrates how individual feature values move the model’s prediction away from the baseline towards the final output, which of course indicates which attributes the model “trusted” for its decision.

The LIME bar plot reinforces this reading by displaying the most prominent contributing features for the same sample in figure 10. It illustrates that features such as num_root = 0 (no processes with root access) significantly nudged the prediction to a normal class, whereas dst_bytes = 0 and logged_in = False nudged the prediction to an attack. SHAP and LIME both always pinpoint logged_in, dst_bytes, service, and protocol_type as important drivers. Applying XAI in such a manner sheds important insight into how the model is able to flag unusual network activities and assists in establishing trust in intrusion detection systems operating in real-world environments. A comparative overview of machine learning model performance and SHAP-based feature importance over the VPN, URL, and IDS datasets is provided in Table 7. The table identifies the top-performing models, most predictive features, and interpretable results from SHAP analysis, focusing on the strength of tree-based models and the importance of domain-specific attributes for cybersecurity classification tasks.

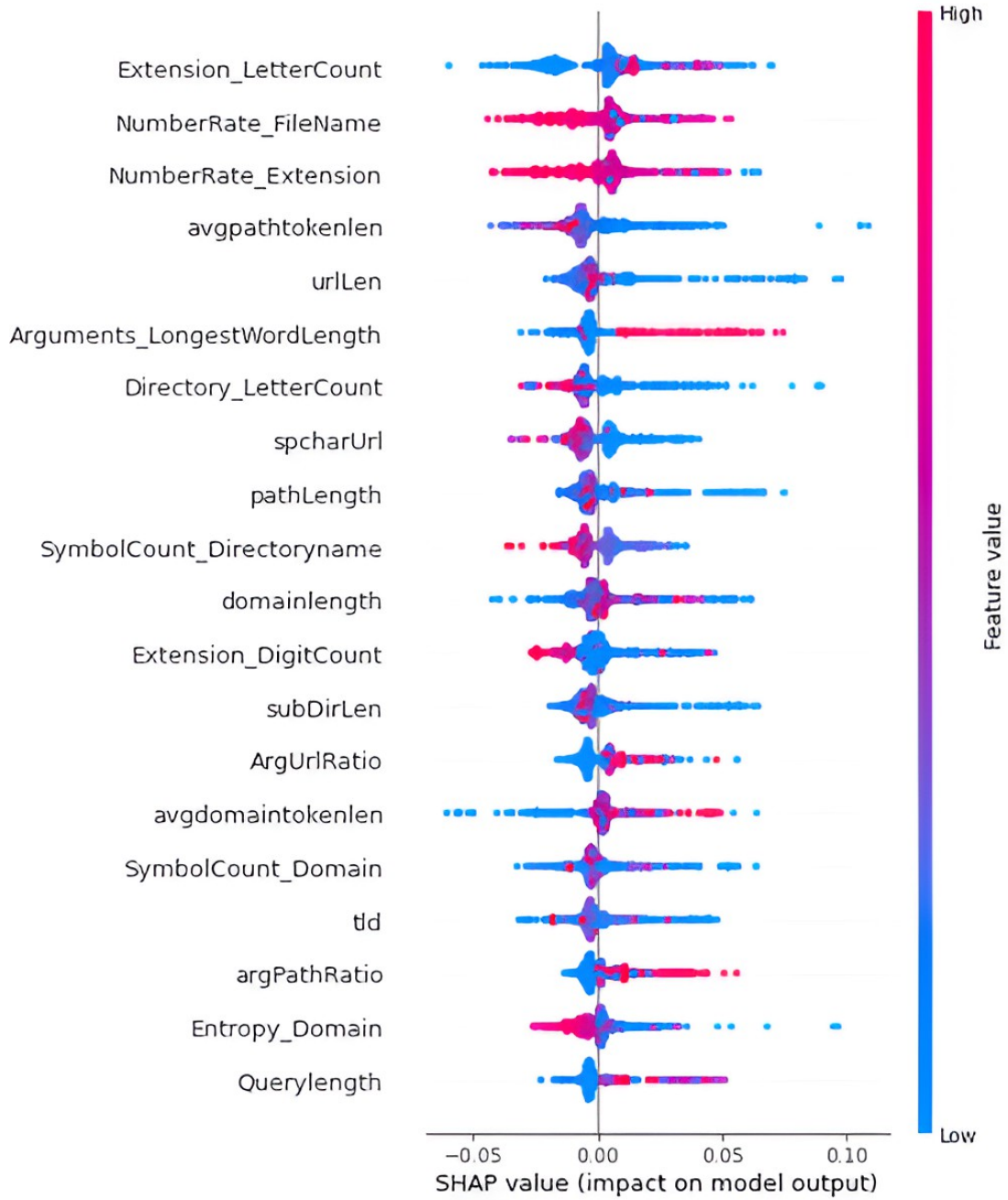


FIGURE 7. Beeswarm Summary Plot of URL traffic

TABLE 7. Summary of Best Performing Models and SHAP Feature Insights across Datasets

Dataset	Best Performing Model	Key SHAP Features	Interpretation
VPN traffic	Random Forest	flowBytesPerSecond, std_flowiat, flowPktsPerSecond	Packet flow statistics are crucial in distinguishing VPN from non-VPN traffic. High or low flow rates significantly impact

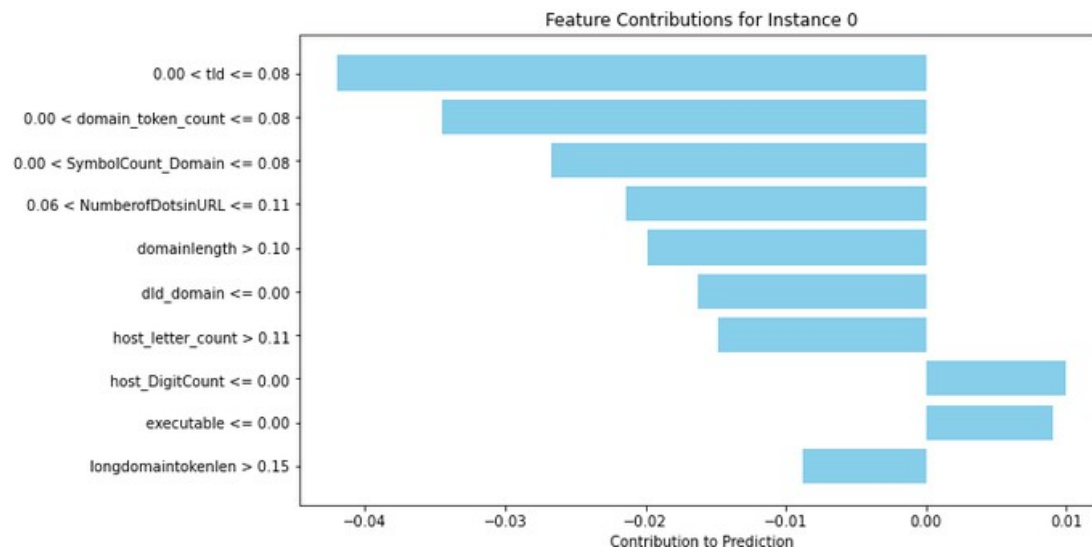


FIGURE 8. Horizontal bar plot of URL traffic



FIGURE 9. Force Plot of IDS dataset

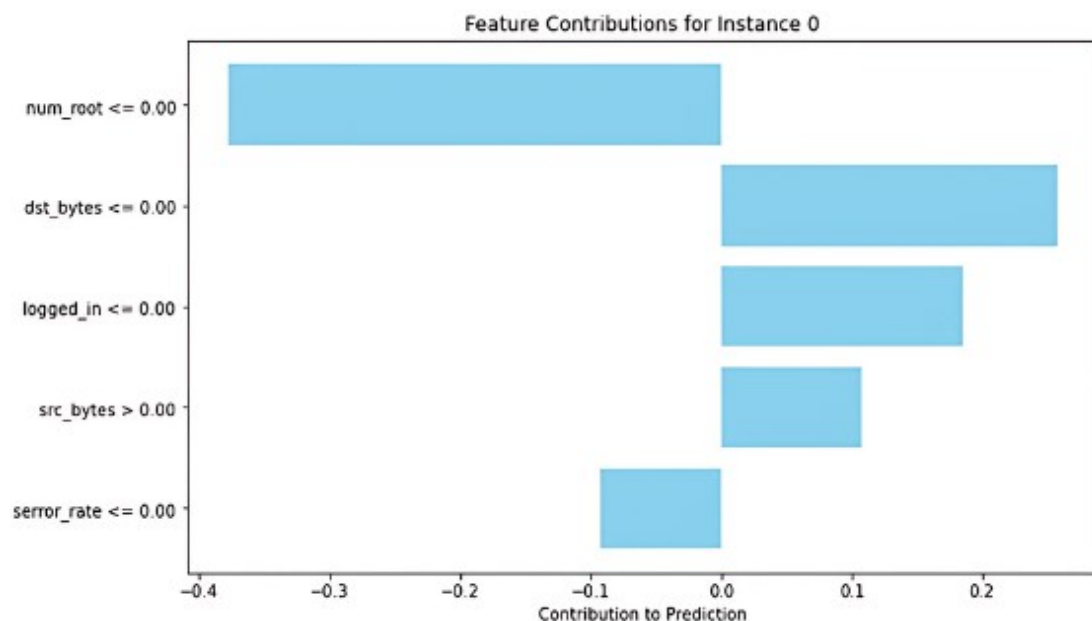


FIGURE 10. Horizontal bar plot of IDS dataset

Table 8 provides a comparative overview of recent studies that integrate Explainable AI (XAI) techniques—specifically SHAP, LIME, and related methods—with machine learning models for network traffic and cybersecurity applications. The table summarizes the datasets, machine learning models, and XAI methods employed in each referenced study, as well as their key findings and interpretability outcomes. By situating our research alongside these selected works, we highlight both the performance of our models and our focus on practitioner-oriented interpretability. The comparison demonstrates that while model performance across studies is broadly comparable, the primary value of our work lies in systematically connecting XAI-derived feature attributions to real-world security behaviors and providing actionable guidance for cybersecurity professionals.

TABLE 8. Comparison with Recent XAI Literature in Cybersecurity

Study / Reference	Dataset(s)	ML Model(s)	XAI Method(s)	Key Results / Findings
Bhupathi et al. (2024) [4]	Phishing URLs	RF, DT, LR, SVM	SHAP, LIME	RF achieved 98.55% accuracy; SHAP/LIME boosted user trust; mapped global/local explanations to phishing features, aiding feature engineering.
Do et al. (2024) [5]	IoT (MQTTset, CIC-IDS2017)	Various (supervised, unsupervised)	SHAP, image-based XAI	SHAP found most trustworthy for interpreting IoT anomaly detection models; XAI improved real-world usability.
Ch et al. (2024) [6]	Memory images (malware)	MLP, SVM, LSTM, GRU	LIME	Classifiers with XAI had high malware detection accuracy and good interpretability; noted computational/scalability challenges.
Zilli et al. (2024)[7]	Network traffic matrices	CNN	LIME, Saliency Maps	LIME/Saliency improved feature attribution, enhanced traffic matrix estimation, reduced data requirements, improved prediction error.
Cerasuolo et al. (2024) [8]	Mobile network traffic	Multi-modal NN	SHAP, other XAI	SHAP identified influential features for app/user activity classification; increased reliability and transparency of decisions.
This Study	ISCX-VPN2016, ISCX-URL2016, ISCX-IDS2012	RF, DT, SVM	SHAP, LIME	Accuracy: VPN 90%, URL 97%, IDS 99%; XAI provided global/local feature attribution, mapped findings to security behaviors, and offered practitioner-oriented guidance.

Following are the summary of the findings of our analysis:

- (i) **Domain-Specific Feature Insights:** The SHAP and LIME explanations demonstrated strong correspondence between model reasoning and established network security knowledge. For VPN traffic classification, timing-based attributes such as packet interarrival time and flow duration emerged as the most influential. This outcome is consistent with the behavior of VPN encapsulation, which alters or smooths timing patterns to conceal underlying traffic types. For phishing URL detection, structural and lexical URL attributes—including token count, domain entropy, and occurrence of specific words—held the highest importance values, reflecting how attackers manipulate URL composition to mislead users. In intrusion and botnet cases, larger packet volumes, flow bursts, and connection rates were prominent, aligning with known characteristics of automated attacks and scanning activities.
- (ii) **XAI-Guided Feature Engineering:** The interpretability results highlight how SHAP and LIME can serve as systematic guides for feature engineering. By isolating features that consistently exhibit high influence across datasets, these XAI tools allow the design of reduced yet representative feature sets that maintain high classification accuracy while lowering computational overhead. This advantage is valuable for real-time or embedded security systems, where a smaller feature pool improves processing efficiency and model retraining speed.
- (iii) **Designing Lightweight and Practical Models:** Insights obtained from the XAI analysis provide a basis for constructing minimalist yet effective detectors. Because SHAP and LIME repeatedly identified only a few dominant features across models, lighter models with fewer inputs can be developed without sacrificing predictive performance. Such compact, interpretable architectures are particularly suitable for deployment in resource-constrained environments (e.g., IoT networks or edge computing nodes) where response time and transparency are critical.
- (iv) **Broader Implications and Future Work:** This work demonstrates how XAI outputs extend beyond interpretability to providing actionable directions for network defense. The identified influential features can be translated into rule-based monitoring strategies or adaptive alert mechanisms, effectively linking model knowledge with operational cybersecurity measures. Future studies will focus on validating this XAI-driven workflow using more recent datasets (such as CIC-IDS-2017/2018) and exploring automated feature-selection pipelines informed by SHAP and LIME rankings. These directions aim to achieve simplified, adaptive, and transparent detection systems capable of addressing evolving cyberthreat landscapes.

6. CONCLUSION

This study demonstrated how the integration of SHAP and LIME can substantially enhance the interpretability and transparency of ML models applied to

encrypted network traffic classification. While Random Forest, Decision Tree, and Support Vector Machine classifiers achieved high accuracy in identifying VPN, phishing, and botnet traffic, their combination with explainable AI techniques provided deeper insight into feature contributions at both the global and local levels. SHAP's global importance rankings and local force plots clarified how specific features influenced predictions across the entire dataset and for individual instances. Similarly, LIME produced intuitive local explanations that revealed the importance of particular features in specific model outcomes, making the decision process more transparent and understandable.

By employing these XAI methods, our work extends beyond achieving model performance to ensuring the models are interpretable, accountable, and actionable—especially crucial in cybersecurity applications where trust and transparency are paramount. The feature importance insights offered by SHAP and LIME can guide future efforts in model refinement, feature engineering, and practical decision-making in domains demanding high reliability.

Although this approach validates established findings rather than introducing novel algorithms or datasets, it delivers a clear, reproducible workflow that bridges the gap between model accuracy and the need for understandable, ethical AI. In doing so, it supports greater trust, reliability, and adoption of automated decision-making systems in real-world security environments, and sets the groundwork for further research in interpretable ML for cybersecurity.

REFERENCES

1. A. K. Cherukuri, S. T. Ikram, G. Li, and X. Liu, *Encrypted Network traffic Analysis*, Springer, 2024, doi: [10.1007/978-3-031-62909-9](https://doi.org/10.1007/978-3-031-62909-9).
2. Y. Zhou, S. Zhou, and J. Cui, "An encrypted traffic classification model based on improved Dense Network and Bidirectional Gated Recurrent Unit," 2024 IEEE 12th International Conference on Computer Science and Network Technology (ICCSNT), pp. 979-984, 2024, doi: [10.1109/ICCSNT62291.2024.10776672](https://doi.org/10.1109/ICCSNT62291.2024.10776672).
3. M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic attribution for deep networks," in *Proceedings of the 34th International Conference on Machine Learning*, 2017, pp. 3319-3328.
4. V. P. Bhupathi, M. Desham, and B. Uma Maheswari, "Enhancing Online Safety: Phishing URL Detection Using Machine Learning and Explainable AI," 2024 15th International Conference on Computing Communication and Networking Technologies (ICCCNT), pp. 1-7, 2024, doi: [10.1109/ICCCNT61001.2024.10723976](https://doi.org/10.1109/ICCCNT61001.2024.10723976).
5. U. Do, L. Lahesoo, R. M. Carnier, and K. Fukuda, "Evaluation of XAI algorithms in IoT traffic anomaly detection," 2024 International Conference on Artificial Intelligence in Information and Communication (ICAIIC), pp. 1-7, 2024. [org/10.1109/ICAIIC60209.2024.10463357](https://doi.org/10.1109/ICAIIC60209.2024.10463357).
6. R. Ch, U. Naresh, J. Manoranjini, S. Telang, S. Pallavi, and S. Kiran, "Advancing Malware Detection using Memory Analysis and Explainable AI Approach," in *Proceedings of the Second International Conference on Intelligent Cyber Physical Systems and Internet of Things (ICoICI 2024)*, Hyderabad, India, 2024, pp. 518-522. [10.1109/ICoICI49053.2024.00083](https://doi.org/10.1109/ICoICI49053.2024.00083).
7. C. Zilli, A. Sacco, D. Monaco, O. Okafor, F. Esposito, and G. Marchetto, "Inferring Visibility of Internet traffic Matrices Using eXplainable AI," *Proceedings of the IEEE Network Operations and Management Symposium (NOMS 2024)*, pp. 979-8- 3503-2793-9/24, 31, 2024, doi: [10.1109/NOMS59830.2024.10575173](https://doi.org/10.1109/NOMS59830.2024.10575173).

8. F. Cerasuolo, I. Guarino, V. Spadari, G. Aceto, and A. Pescapé, "XAI for Interpretable Multimodal Architectures with Contextual Input in Mobile Network traffic Classification"
9. S. M. Lundberg and S. I. Lee, "A unified approach to interpreting model predictions," in Proceedings of the 31st International Conference on Neural Information Processing Systems, 2017, pp. 4765–4774, ACM Digital Library. [10.5555/3295222.3295230](#)
10. M. T. Ribeiro, S. Singh, and C. Guestrin, "Why should I trust you? Explaining the predictions of any classifier," in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2016, pp. 1135–1144, ACM Digital Library. [10.1145/2939672.2939778](#)
11. S. Mali, M. Gujral, and A. Cherukuri, "Encrypted Network traffic Classification Using Intelligent Techniques," Cureus Journal of Computer Science, vol. 2, Article es44389-024-02701-2, Jan. 16, 2025, doi: [10.7759/s44389-024-02701-2](#).
12. Jonnalagadda, A., Mohanty, D., Zakee, A., & Kamalov, F. (2024). Modelling data poisoning attacks against convolutional neural networks. Journal of Information Knowledge Management, 23(02), 2450022. [10.1142/S0219649224500229](#)

¹ SCHOOL OF COMPUTER SCIENCE ENGINEERING AND INFORMATION SYSTEMS, VELLORE INSTITUTE OF TECHNOLOGY, VELLORE, INDIA.

Email address: cherukuri@acm.org