Section A-Research paper
ISSN 2063-5346

OPTIMAL FEATURE PICKING FOR INTRUSION DETECTION ON THE BASIS OF EXPLAINABLE ARTIFICIAL INTELLIGENCE

[1]Mr. Thamraj Narendra Ghorsad, [2] Dr. Amol V. Zade
[1]Ph.D. Scholar, Computer Science & Engineering Department, G. H. Raisoni University Amravati,
[2]Professor, Computer Science & Engineering Department, G. H. Raisoni University Amravati,
[1]raj.ghorsad@gmail.com, [2]amol.zade@ghru.edu.in

Abstract: Traditional security mechanisms for intrusion detection are strong, and the use of machine learning algorithms to detectintrusion into IoT networks has increased. As a result, the efficiency of the IoT network has increased and an effective model can be developed to detect attacks on the network. UNSW-NB15 is an IoT-based publicly available dataset for traffic data that contains general activity and malicious activity. Using this dataset, the key features were selected and these features fed different classifiers for training and classifying the attack in the network. Ann model performance levels of accuracy, the ML classifier, and accompanying algorithms for constructing a digital security system based on packet flow in-network, and features that really can monitor anomalous botnet behaviors proven to be highly efficient. The decision-making model for the classification was visualized using proven Explainable AI (XAI) approaches utilizing Scikit-Learn, LIME, ELI5, and SHAP libraries to boost explainability in classification predictions. The findings showed that XAI is realistic and profitable since security expertise and specialists stand to benefit greatly from combining conventional machine learning methods with Explainable AI (XAI) methodologies.

Keywords: Intrusion Detection Systems, Machine Learning, Explainable AI, Decision Trees, Multi-Layer Perceptrons, and XGBoost

1. Introduction

Because of the enormous number of important user information that IoT systems handle, become more important targets for hostile cyber attacks [1]. It's especially important for vital operations because intruders try to breach cyber security strategies including privacy, authenticity, and reliability. In [2], suggested that today's age is the age of competition, which has led to increased competition in every business, using efficient and productive big data and powerful technology. Such businesses need to be protected from sensitive data theft and network attacks. There may be some similar features between the data in the social network and the malicious traffic in the network for intrusion detection. It is necessary to create an efficient and effective model to discover such hidden important features and knowledge [3]. With the increasing use of the Internet and the advent of new features, a strong security system is needed to address potential threats such as attacks on network systems [4]. In response, network intrusion detection systems (IDS) or systems that monitor and detect cyber-attack patterns over networking environments based on machine learning algorithms/models have been developed for detecting abnormal activity in the network where they monitor network traffic for suspicious activities and issue alert in case of detected attack types [5]. Machine Learning (ML) algorithms are being investigated as potential IDS frameworks as current or traditional IDS capabilities have concerns including:

- Because of the variations in IoT networks and behavior, conventional network security mechanisms could not beappropriate.
- Low computational capability and efficient use of consumption.
- There is no unified benchmark for IoT structure, guidelines, and connection areas, thus conventional security measures such as data encryption and authorization can be simply used.

To detect malicious activity or botnets, existing machine learning algorithms and models can learn IoT network inputs related to target features relating to Normal or Attack behaviors. However, current research suggests that these ML-based IDSs are primarily "black boxes," with users of cyber security services lacking the ability to explain how the system arrived at the attack prediction or classification, which is critical for an optimal initial evaluation in cyber security and information security planning and resource allocation for IoT networks. The fast integration of IoT networks in a wide range of applications and settings has increased demandfor successful security practices and systems, but relevant researchers, security administrators, information security professionals, and others would greatly benefit from increased capabilities in understanding the ML-based IDS systems they may use to conduct operations and develop computer or network protection strategies to protect assets.

The objective of this work proposes to apply a survey of ML algorithms that have been modified and considered Explainable AI (XAI) methods through the utilization of existing Python libraries to explain model classification decisions, the logic behind predictions, and feature importance for predictability to increase the transparency of ML-based IDSs to be understood further than current levels by human analysts in the IoT cyber security domain. These tools have been utilized in varying applications and showpromise to extend explainability features [6] to IoT network security IDS problems. Additionally, performance metrics of accuracywill be measured to augment explainability.

Related Work

There are some related to the ELI%, LIME and SHAP. There are very few previously work done on ELI5, LIME, and SHAP in the field of IDS. In [7]-[8] suggested the image visualization model based on LIME and natural language processing for image classification. In [9]-[10] utilized the SHAP for extracting the content music for analysis and black-box prediction [23]. This approach does not directly use for intrusion detection.

In last few decades, machine learning and deep learning techniques performed well in the field of intrusion detection. As outcome most of researcher used deep learning approach for attack detection and bahaviour analysis in IoT network. Protecting Internet of Things (IoT) networks has been a critical area of research for cyber security experts as IoT continues to be integrated into applications including Transport systems, remote monitoring, advanced healthcare services, and advanced manufacturing. Furthermore, developed Intrusion detection systems based on ML techniques and other statistical feature learning algorithms have been researched and put into application minimally. In [11], suggested the machine learning model for intrusion detection based on the extraction of a statistical feature on traffic data in an IoT environment. The statistical feature extraction was used to analyze the network traffic. The various machine learning classifier was used for detecting the malicious activity in the network over the UNSW-NB15 and NIMS botnet datasets.

In [12], suggested the machine learning model for intrusion detection over the IoT environment and classifying the attacks using the sensor data. The various sensor was used for data acquisition and processed by Node MCU ESP8266. The performance evaluatedon Man in the Middle attack using ARP. In [13], suggested the various approach to deep learning for detection of intrusion over theIoT network and implementation the state-of-art machine learning algorithms for attack classification. Many other machine learningtechniques used for intrusion detection have been researched and developed as outlined. This research investigates the range of machine learning approaches used over the IoT network for Intrusion

detection. Over 95 works on the subject were analyzed, whichranged across different sub-disciplines in machine learning and security issues in IoT environments [14].

The effectiveness of the suggested model was the subject of the majority of IDS-based investigations. In [15] looked into a numberof techniques, including k-means clustering, Naive Bayes, SVM, and OneR techniques. The effectiveness of this model was improved for both the normal flow of data and DoS attacks. In addition, a genetic algorithm was used to improve the identificationof various forms of assaults with a 97 percent efficiency [16].

The author of [17] concentrated on many forms of threats, including http flood, smurf, siddos, and UDP flood. They used a varietyof machine learning techniques to identify DoS attacks, with an accuracy rate of 98% utilizing an MLP technique. To increase detection performance [18] suggested a decision tree model for intrusion detection. This approach outperformed both the Naive Bayes and the KNN techniques. The investigators were continuously looking for a good technique to identify attacks that are high- performing, fast, and have a false alarm [19]-[20]. The IDS mechanism is primarily concerned with reliability. The easily understoodaspect of prediction techniques receives less attention.

In their paper, they developed a deep neural network for network intrusion detection and proposed an explainable AI framework to demonstrate model transparency throughout the machine learning pipeline. Utilizing existing XAI algorithms generated from SHAP, LIME, Contrastive Explanations Method (CEM), Protidic, and Boolean Decision Rules via Column Generation (BRCG), that provide explanations on individual predictions, they applied the approaches to the NSL-KDD dataset demonstrating a successful increase in model transparency.

1. Overview & Benefits of Explainable (XAI) Machine Learning

Advanced and state-of-the-art ML algorithms and models offer valuable applications in establishing better IoT network security. The ML techniques, learn from input features generated in network traffic and offer support to cyber security personnel in making critical threat detection decisions. However, these techniques are based on advanced models that are too complex to be interpreted by human analysts; hence, may they turn to traditional tools that may not be as viable but offer more explainability or inherent trust by the human involved. In many cases, it is nearly impossible to get a feeling for the inner workings of an ML system for Intrusion Detection. This may further decrease trust that a certain prediction from the model is correct even though performance results may indicate otherwise. Having an intuitive explanation of the rationale behind individual predictions or model decision-making framework will better position cyber security experts to trust prediction or the classifiers themselves, especially, in understanding how it behaves in particular cases. Explainable AI (XAI) offers a variety of explanation or feature importance tools for generating explanations about the knowledge captured by trained ML models to aid in increasing overall trust.

2. Proposed Methodology

This section introduces the proposed model for the selection of key features used to improve IDS on IoT networks. Figure 1 shows the overall structure of the proposed model. The proposed model consists of three parts, the first part (on the left) showsthe traditional approach to IDS and the second part (center) shows the adoption of machine Python library models for the selection of key features, and the third part to get important features. Finally, the machine learning classifiers get the estimatedresults. The traditional approach standard UNSW-NB15 dataset is used, trained the proposed model using four machine learning classifiers namely Decision Tree, Multi-layer Perceptron, and XGBoost. All the classifiers integrate with the proposedExplainable AI classifier for improving the decisions, and interpretability for IDS. All the classifiers' results are compared with each other. The aim of this study is to enhance the interpretability of intrusion detection over the IoT network. In this paper ELI5, LIME, and SHAP python library are introduced for feature selection. This approach also determines intrusion detectionprediction and gives significant features.

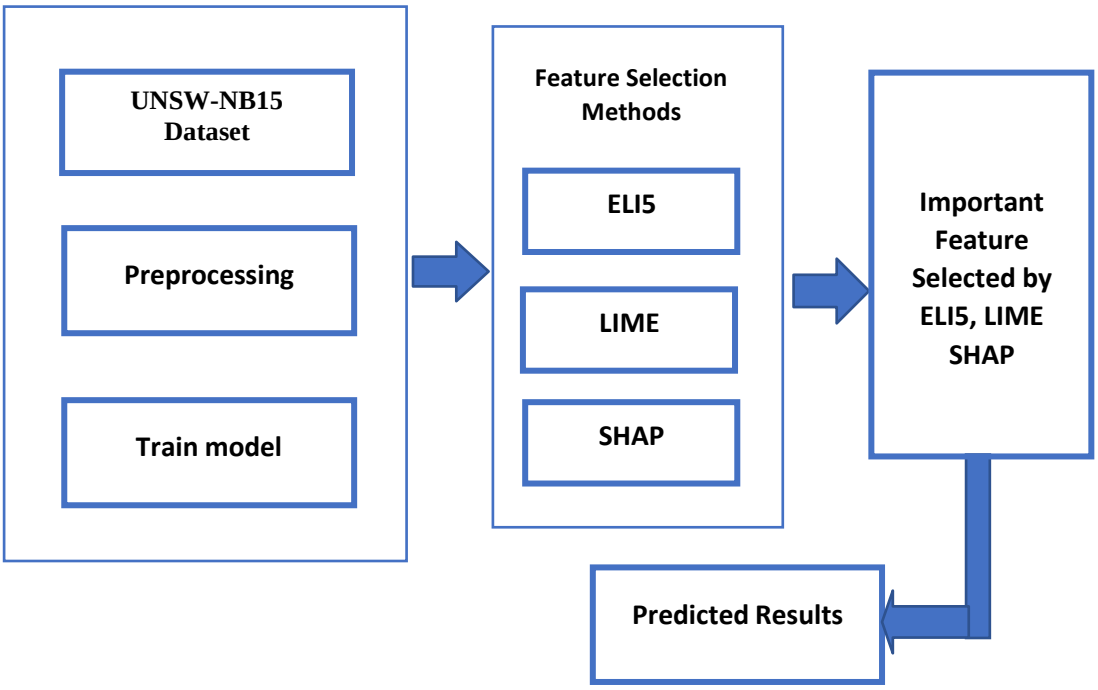


Figure 1: Structure of Proposed System for feature selection of IDS

UNSW-NB15 Dataset

UNSW-NB15 is a publically available benchmark dataset that contains network traffic data based on the Internet of Things environment, including multiple subcategories for regular operations and harmful attack behaviors from botnets (through classification of attack types including Fuzzers, Analysis, Backdoors, DoS, Exploits, Generic, Reconnaissance, Shellcode, and Worms). This benchmark dataset is developed from the Australian Centre for Cyber Security's (ACCS) Cyber Range Lab which contains normal activity data and malicious activity data of the IoT network [21]. The proposed model is based on the IXIA trafficgenerator architecture as shown in figure 2 and source of this architecture is cited.

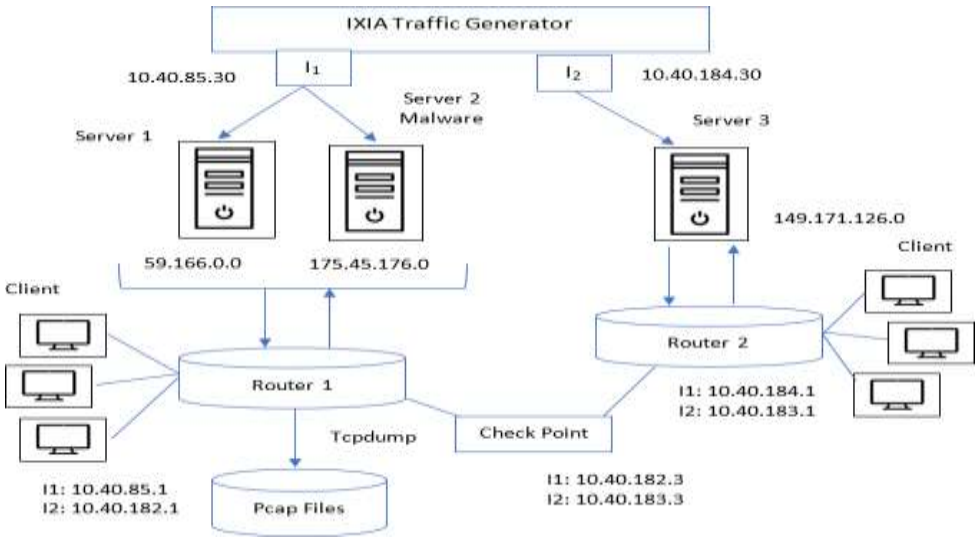


Figure 2: IXIA Traffic Generator Overview [22]

The UNSW-NB15 is pre-partitioned by its creators into being configured into a training set for model training and a testing set for model performance. This benchmark dataset contains 175,341 records for training the model. For the testing 82,332 records are available with a target response of the traffic behavior for each record, attack, and normal behavior. The dataset consists of 39 numeric features. The features and their descriptions are listed in the UNSW-NB15_features.csv file.

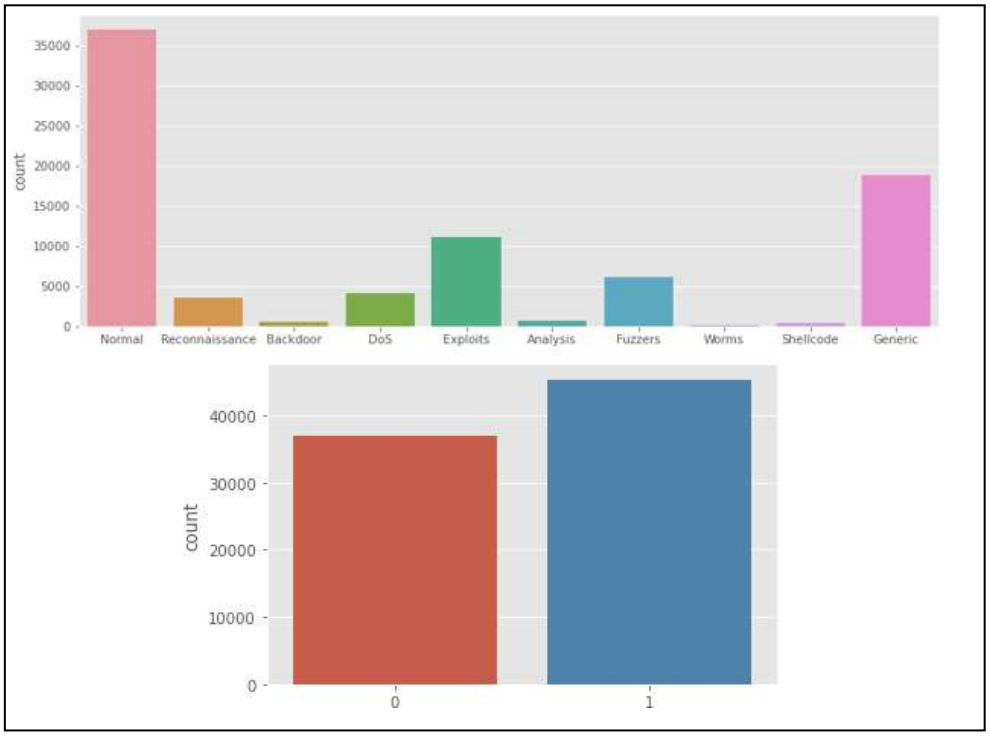


Figure 3: Training Dataset Distribution and Counts

For our experimental processes, the target feature will be a binary classification of Normal or Attack behavior. Figures 3 provide the details and the values distribution of each attack class within the data subsets, where 0 represents Normal and 1 represents Attack behavior. We could see that the dataset is adequately balanced for the binary response variable of activity behavior.

The three supervised ML approaches that will be used to develop binary classification classifiers are Decision Trees, Neural networks based on Multi-layer Perceptron, and XGBoost. The ML algorithms in the mentioned order offer decreasing capabilities for explainability (XAI).

Decision Tree

The decision tree (DT) classifier is a supervised ML algorithm that will be utilized for the classification task of Normal or Attack behavior based on the 39-input feature. The resulting DT algorithm develops a decision-making process based on a tree-like model with nodes or branches. The max depth of the decision tree can be defined beforehand. A decision tree is already an explainable machine learning algorithm through visualizations of the resulting trees.

Multi-layer Perceptron

Multi-Layer Perceptron (MLP) classifiers are artificial neural networks that utilize perceptrons or single neuron models for complex predictive modeling tasks. The MLP classifier as a neural net has an inherent ability to learn the representation in your training data and how to best relate it to the output variable that you want to predict.

In this sense, neural networks learn mapping. The building block for neural networks is artificial neurons, which weighted input signals and produce an output signal using an activation function. The activation function is a mapping of summed weighted input to the output of the neuron and contains the threshold at which the neuron is activated and the strength of the output signal. Neurons are arranged into layers of neurons, and layers create a network, where the weights initially are best guesses. The most preferred way training algorithms for neural networks are gradient descent or back-propagation algorithm with sigmoid function, in which the network processes the input upward activating neuronsas it goes to finally produce an output value. The network's result is given to what is predicted, and the error rate is determined. The error is then transmitted down throughout the network, one level at a time, and the values of the weights are adjusted based on their contribution to the error. Like the weights, convergence to a locally optimal, the operation is continued for all instances in the dataset.

Neural networks like MLP Classifiers, for the most part, lack sufficient model explainability and interpretability. In the tradeoff between the explainability/interpretability of an algorithm and its accuracy in the application, neural networks heavily lean more toward prediction performance. Neural networks contain visible layers and hidden layers of neural units, which hidden layers and their unknown interaction post-training significantly cause neural networks to act as "black-box" algorithms instead.

XGBoost Classifier

Boosting is an ensemble strategy that contains various models are included to fix old models' flaws. Gradient boosting is a technique that involves creating novel versions of ML models that forecast the errors of previous ML models and then combined them to form the final result. In addition, while introducing additional models, the gradient descent technique reduces the error rate. This methodwill aid in the prediction and classification of Normal and Attack behavior in an IoT environment.

Proposed Approach with Scikit-learn, XGBoost, and XAI Libraries

UNSW-NB15 training dataset after applying data processing techniques for data cleaning, normalization, and transformation will be used to train each of the three supervised ML binary classifiers: Decision Trees, Neural Network based on Multi-layer Perceptron,and XGBoost. The target feature will be a binary classification of Normal (0) or Attack (1) behavior. Thereafter, the next process will be to test the trained model using the data processed UNSW-NB15 testing dataset. The model performance will be evaluated using the accuracy score. The procedure described above is will not be tuned using model or classifier hyper parameters. Scikit- Learn implementation of the Decision Trees Classifier and Multi-layer Perceptron Classifier will be utilized, while the XGBoost library will be utilized for the XGBoost Classifier. After classifiers are trained and tested, the next process is to develop interpretable diagrams, feature importance plots, and classification/prediction explanation visuals based on the trained classifiers used to detect network traffic behavior in the testing set.

1. ELI5 is a visualization python library that is useful for debugging machine learning models and explaining the predictions theyhave produced. It used to select the important features from the database. following equation shows the ELI5.

$y = c + m_1x_1 + m_2 2 + \ldots . + m_n x_n$ (1)where x is the independent variable and m is the coefficient

2. LIME is called as local interpretable model-agnostic explanations is technique for predicting the feature from dataset by machinelearning algorithms. The following equations specify the LIME.

$$\emptyset(x) = argmin_{p \in G} L(f, g, \pi_x) + \Omega(p)$$

Where, G is the interpretable class models.

SHAP (SHapley Additive exPlanations) is a game-theoretic approach to explaining the output of any machine learning model.SHAP to help better understand the impact of features on the model output. The following equation specify the SHAP [24].

$$(z') = \emptyset_0 + \sum^{N}{}_{i=1} \emptyset_i z_i' \quad (2)$$

Where, p is the exPlanations model, $z' \in \{0, 1\}$ is coalition vector. N is the maximum size of the coalition. $\emptyset_i \in R$ is the importantattribute of feature i in dataset. 1 indicate the feature are present and 0 indicate the absent. For every instance of x, the coalition vector for all 1s is present using the given equation 3.

$$(z') = \emptyset_0 + \sum^{N}{}_{i=1} \emptyset_i \quad (3)$$

4. Feature Extraction

Decision Tree Classifier

Using the Scikit-learn library's tree. DecisionTreeClassifier (), the training set was used to build a Decision Tree classification model for Normal or Attack behavior. The model performance accuracy against the testing set was 85% as indicated in Figure 4. The importance of the top 10 features was graphed with the sci-kit-learn library and ELI5's Permutation Importance toolkit. The important features are measured as reducing the weight impurity of the node in the network by using the node probability. The mostimportant features will be higher in the tree-like visualization generated.

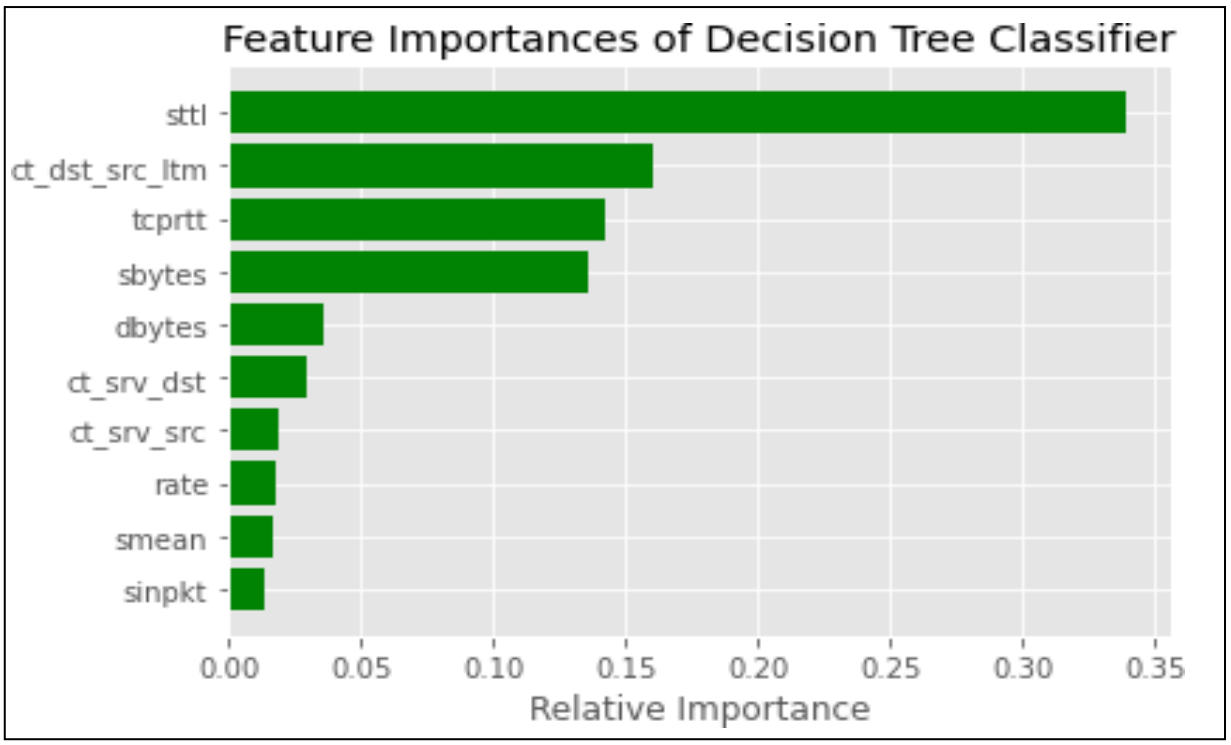
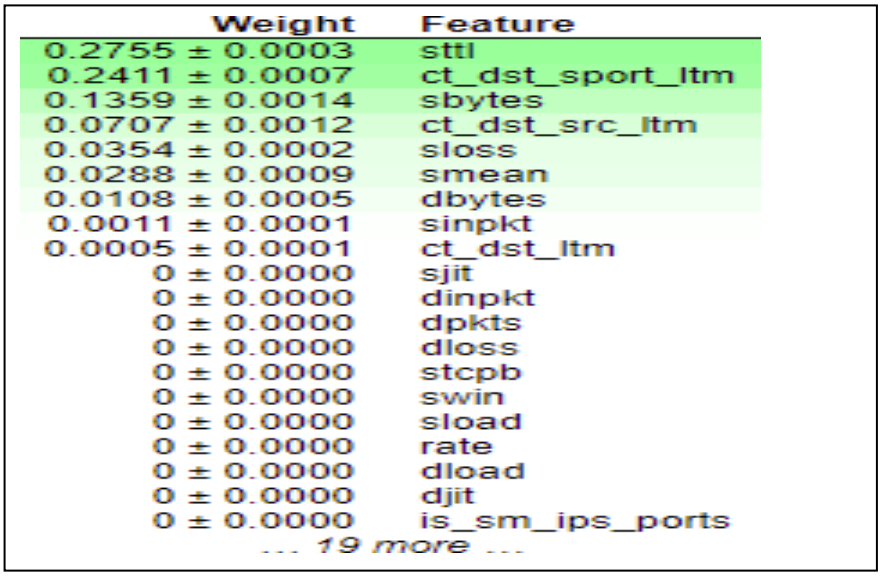


Figure 4: Top feature by ELI5 using Decision: Scikit Learn



Weight	Feature
0.2755 ± 0.0003	sttl
0.2411 ± 0.0007	ct_dst_sport_ltm
0.1359 ± 0.0014	sbytes
0.0707 ± 0.0012	ct_dst_src_ltm
0.0354 ± 0.0002	sloss
0.0288 ± 0.0009	smean
0.0108 ± 0.0005	dbytes
0.0011 ± 0.0001	sinpkt
0.0005 ± 0.0001	ct_dst_ltm
0 ± 0.0000	sjit
0 ± 0.0000	dinpkt
0 ± 0.0000	dpkts
0 ± 0.0000	dloss
0 ± 0.0000	stcpb
0 ± 0.0000	swin
0 ± 0.0000	sload
0 ± 0.0000	rate
0 ± 0.0000	dload
0 ± 0.0000	djit
0 ± 0.0000	is_sm_ips_ports
… 19 more …	

Figure 5: Decision Tree Feature Importance: ELI5 Permutation Importance

Both of the feature importance outputs indicate very similar results with feature 'sttl' or "source to destination time to live value" in the network traffic analysis being indicated as the most important to classification prediction. The most important features can bevisualized in the upper layers of the decision tree visualization in Figures 6 through 7. Let us visualize the first three levels of the decision tree, max_depth=3, 5, 8 as shown in figure 6, figure 7, and figure 8.

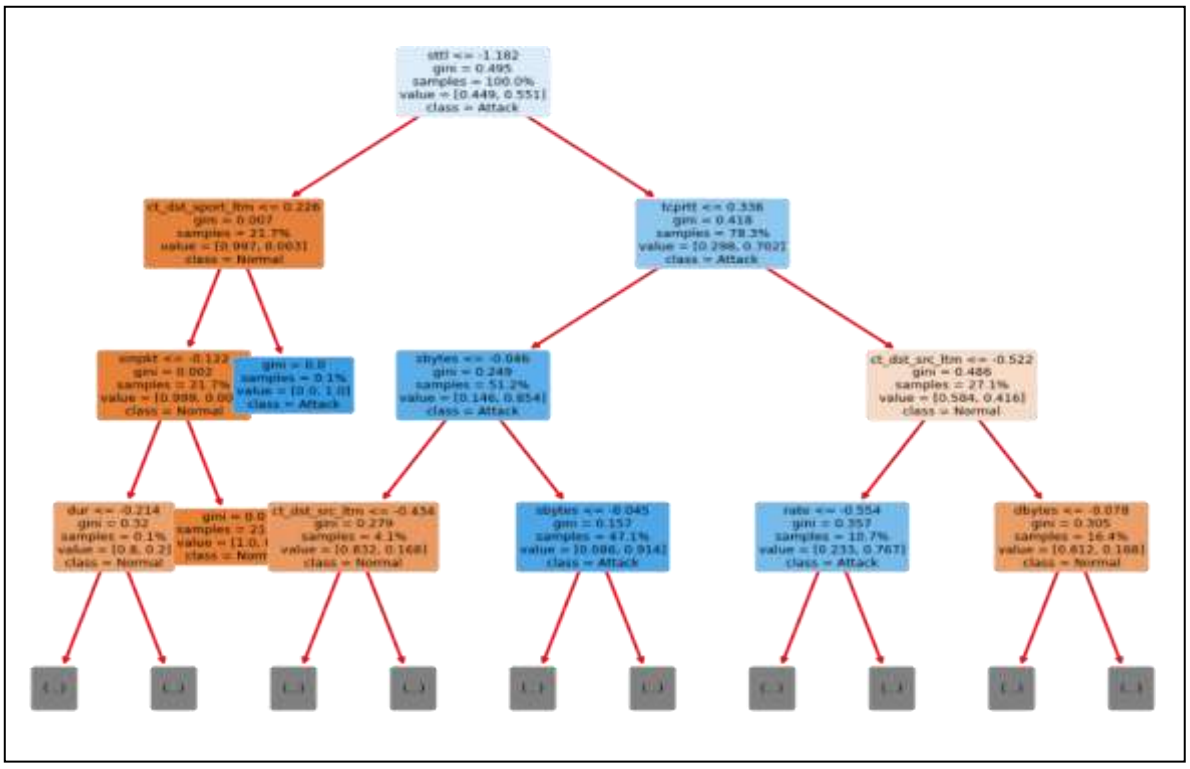


Figure 6: Decision Tree Classifier (Depth = 3 Nodes) Explainable AI Visualization

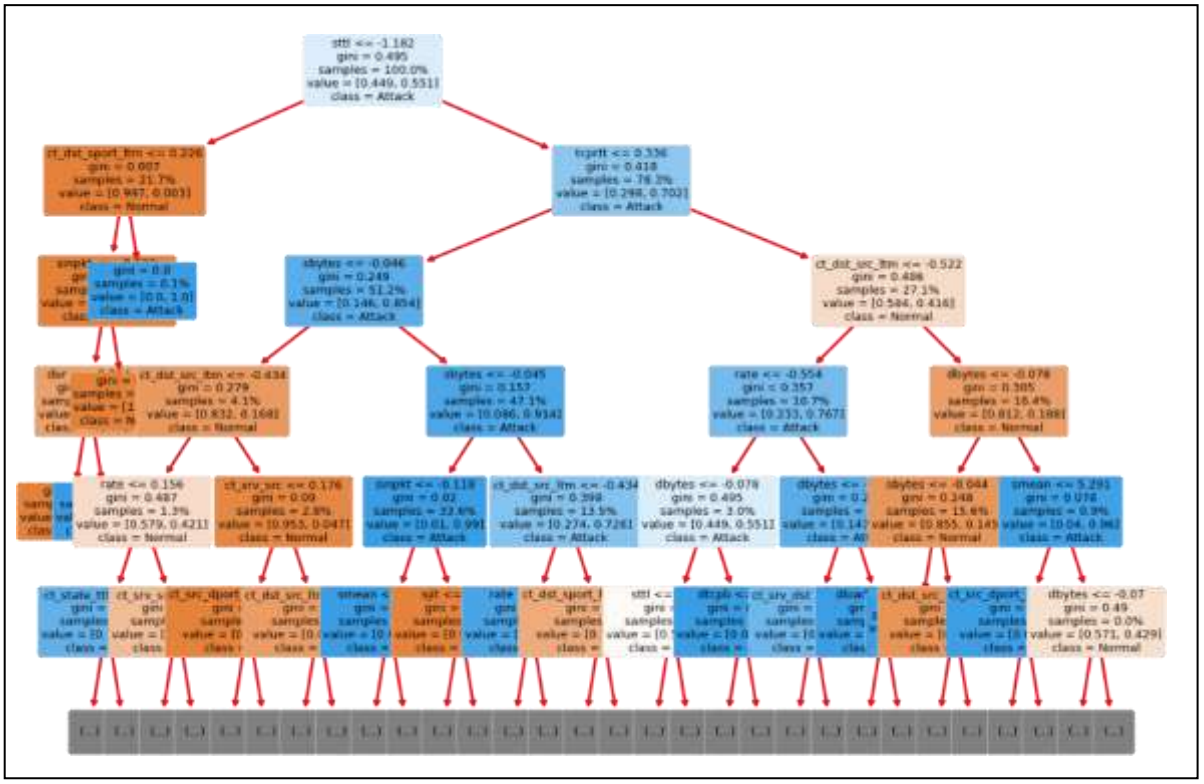


Figure 7: Decision Tree Classifier (Depth = 5 Nodes) Explainable AI Visualization

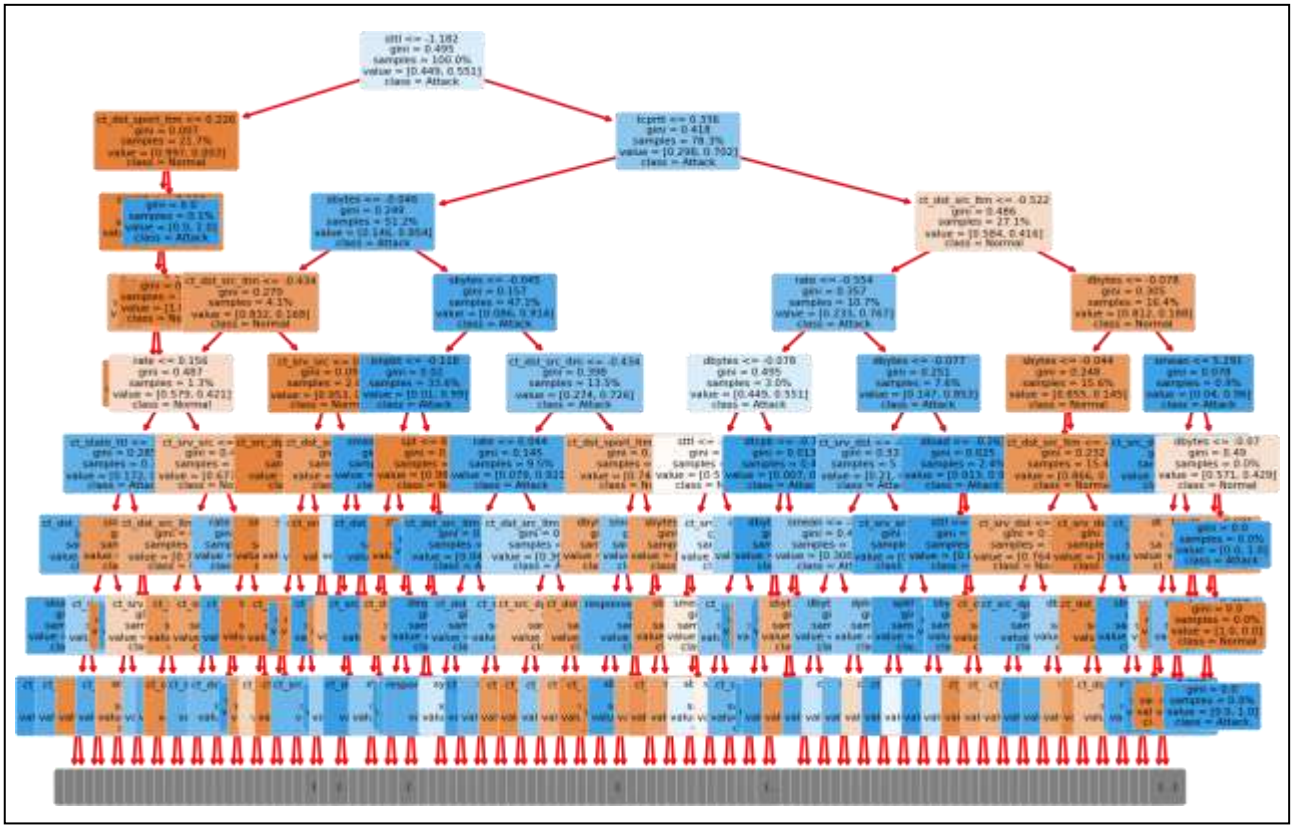


Figure 8: Decision Tree Classifier (Depth = 8 Nodes) Explainable AI Visualization

The decision tree visualizations enable the model to explain the ability through inspection of each decision level and its associated feature and splitting value for each condition. If a certain network traffic sample satisfies the condition, it goes to the left branch or node, otherwise, it goes to the right branch. Additionally, in each class line, the classification prediction result is depicted dependingon the max depth of the tree selected. Utilizing decision trees for IoT network traffic ML-based IDSs provide high accuracy classification results, indicating robust detection of malicious threats. Furthermore, the explainability features of the DT algorithm based on plotting decision trees can help human analysts understand the model. This will allow for a greater understanding of the cybersecurity landscape around IoT networks. This understanding includes theorizing what the IDS machine-learned from the features or comparing expectations. The human analysts may further aid the machine in learning through adding features or feature engineering using domain knowledge. This will significantly help analysts assess the correctness of the model decision framework and improve upon it.

Multi-layer Perceptron (MLP) Classifier

For the MLP classifier, the model was trained and tested on the corresponding datasets. The overall model performance accuracy against the testing set was 89.83%. This indicates a very exceptional classification prediction score for detecting Normal or Attack behavior in IoT traffic. Using the LIME - Local Interpretable Model-Agnostic Explanations library, a model prediction visual of the MLP Classifier can be generated for individual predictions in the training set. LIME perturbs the original data features and prediction, to feed into a developed internal classification model, and then observes the outputs. Thereafter, the library weighs the new data outputs as a function of their proximity to the original point. Next, it fits a surrogate linear regression on the dataset with variations using the sample weights. Lastly, the original data points can be explained with the newly trained explanation model. Figure 9 displays an example of the Lime Tabular Explainer output with the top 5 features indicated.

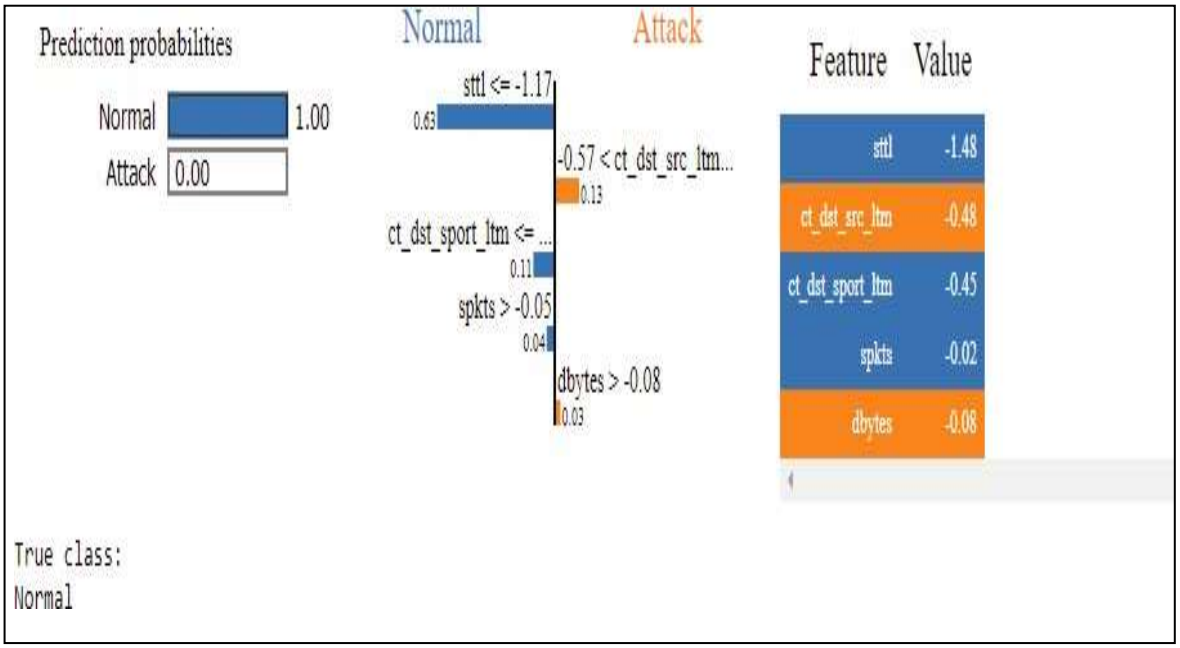


Figure 9: Single Classification Prediction using the MLP Classifier Explanation

The visual dashboard indicates which features and their weights brought the overall behavior classification to be predicted as Normal for that network traffic record. This classification is inspected to be correct as the true class is 'Normal'. This visual dashboard offers robust individual explainability of predicted classifications. Human analysts can conduct in-depth analysis for cyber securityresearch or follow-up assessment on why certain network traffic was classified in which they were by the model. This tool offers increased transparency capability of predictions that can be exploited for future cyber security research while utilizing the high- performance benefits of a neural net MLP classifier, which are functionally 'black boxes.

XGBoost Classifier

Similarly, to the other two classifiers, the XGBoost Classifier was trained for the classification task and tested on the testing set. The overall model performance accuracy was 89.89%, demonstrating the high capability of the XGBoost classifier to classify network behavior. The performance is approximately similar to the MLP Classifier.

The SHAP (SHapley Additive exPlanations) library was utilized to utilize explainability capabilities with this classifier. The SHAPlibrary offers the ability to analyze which training samples and features to determine a better impact on ML model results. SHAP's main advantages are local explanation and consistency in tree-based model structures such as XGBoost. SHAP creates values that interpret results from tree-based models. It is based on value calculations from game theory and provides extensive feature importance using 'marginal contribution to the results of the model.

To explain predictions, the Tree SHAP implementation integrated into XGBoost can be used to explain the testing set classificationpredictions. Figure 10 provides a visualization into explaining a single prediction, while Figure 11 captures an explanation of many predictions through feature comparison or output classification values. The (x) values provide a classification value, where closer to 1 indicates Attack behavior, while closer to 0 indicates Normal Activity by a network traffic record.

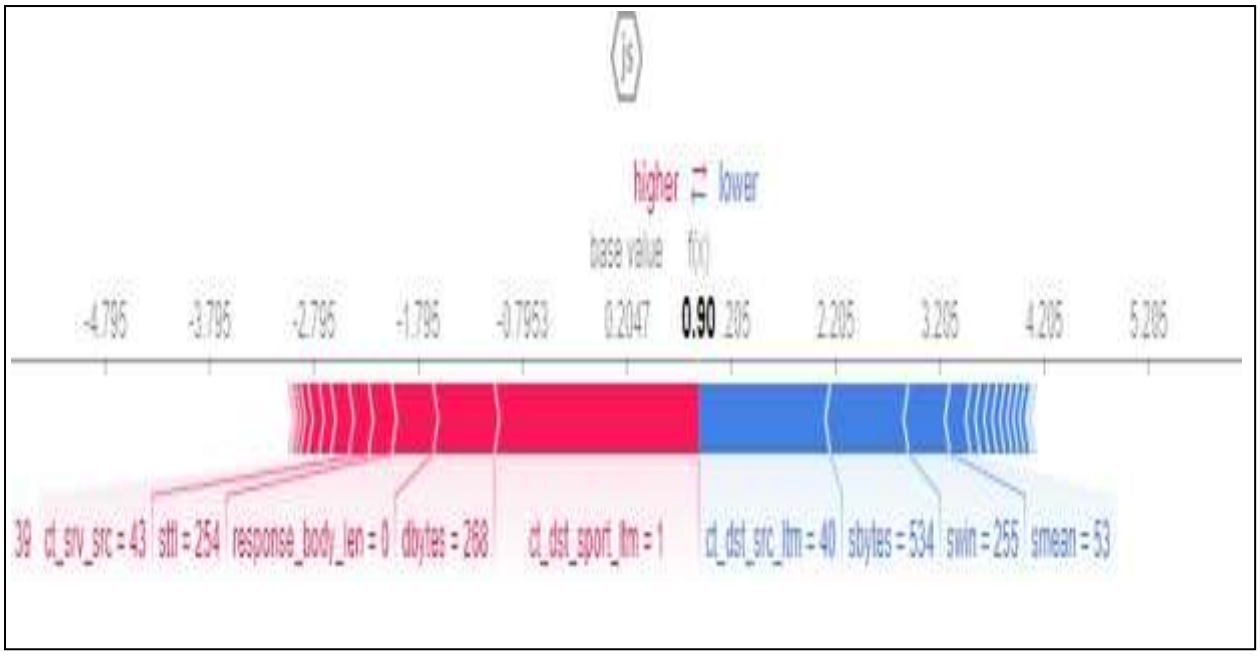


Figure 10: XGBoost SHAP - Visualize many predictions

A feature importance plot through SHAP is conveyed in Figure 12 to determine the mean importance of input training features to predict classification. The results are similar to the DT Classifier.

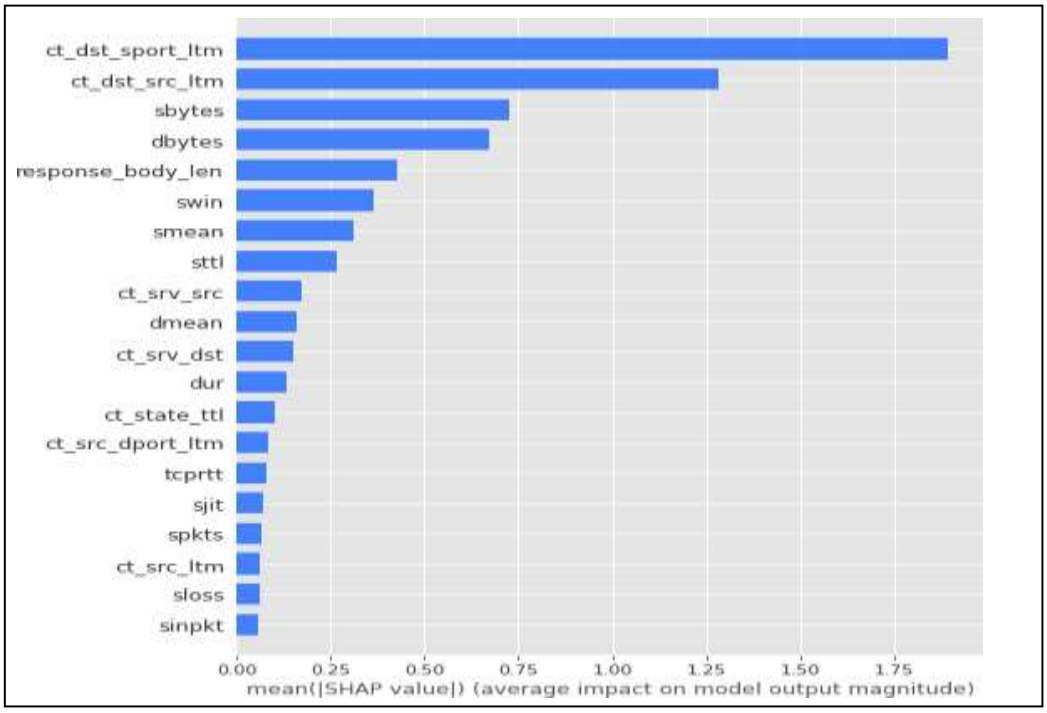


Figure 11: Top feature by SHAP on XGBoost Classifier

The SHAP summary plot shows the top feature combinations while providing visual indicators of how feature values affect classification predictions. In Figure 12, red indicates a higher feature value, and blue indicates a lower feature value. On the x-axis, a higher SHAP value to the right corresponds to the prediction value (Attack Behavior), lower SHAP value to the left corresponds to a lower prediction value (Normal Activity).

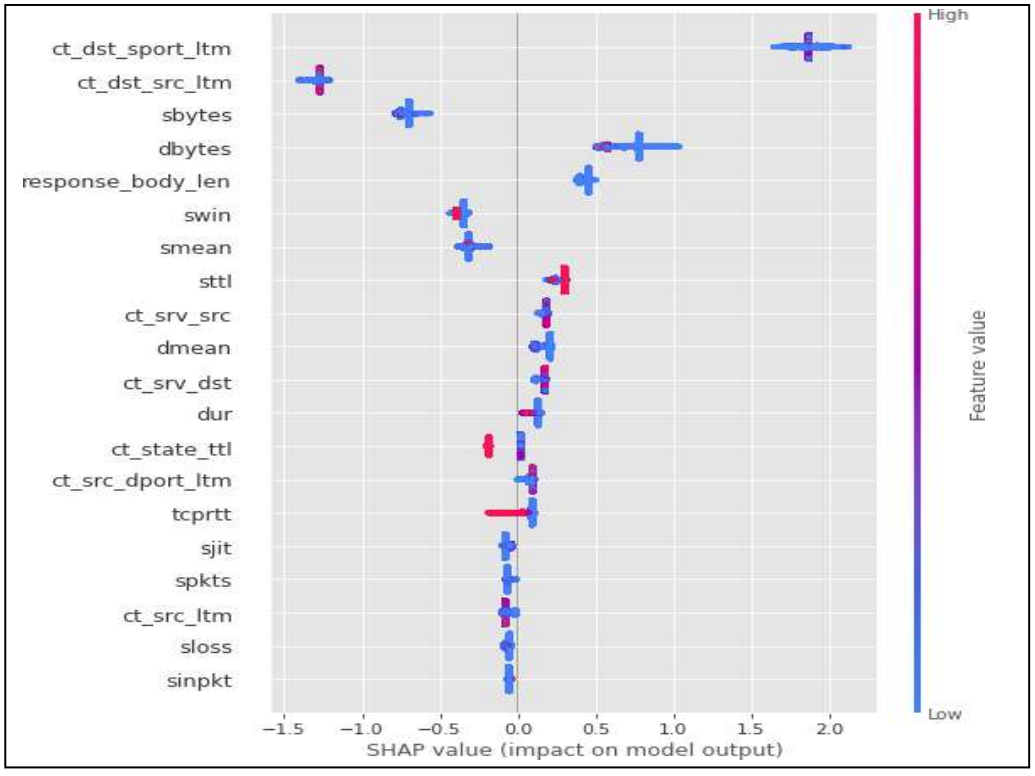


Figure 12: SHAP Summary Plot for XGBoost

Figure 13 shows the dependence plots SHAP values which reflect the effect of a single feature on the entire database. The dependence plot shows the feature's value vs. the SHAP value for the functional relationship that exists in the features of datasets.

Furthermore, the SHAP values can create SHAP dependence plots, which show the effect of a single feature across the whole dataset. They plot a feature's value vs. the SHAP value of that feature across many samples and account for interaction effects present in the features. A matrix of overview graphs is plotted with the major impacts on the horizontal and the combined impact ofthe diagonal in an overview graph of a SHAP interactions value matrix.

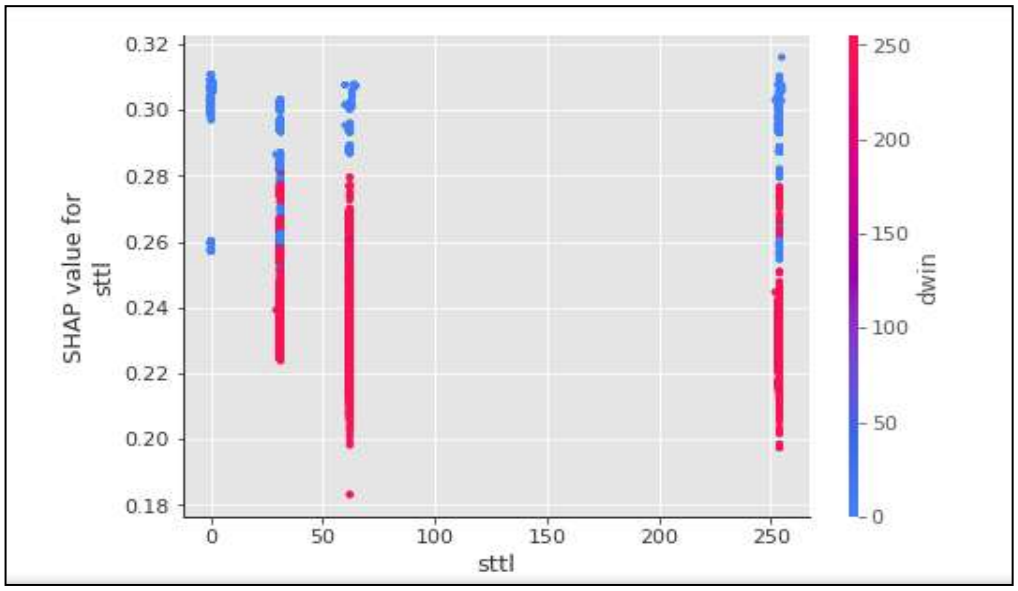


Figure 13: SHAP Dependence Plots for ‘sttl’ feature

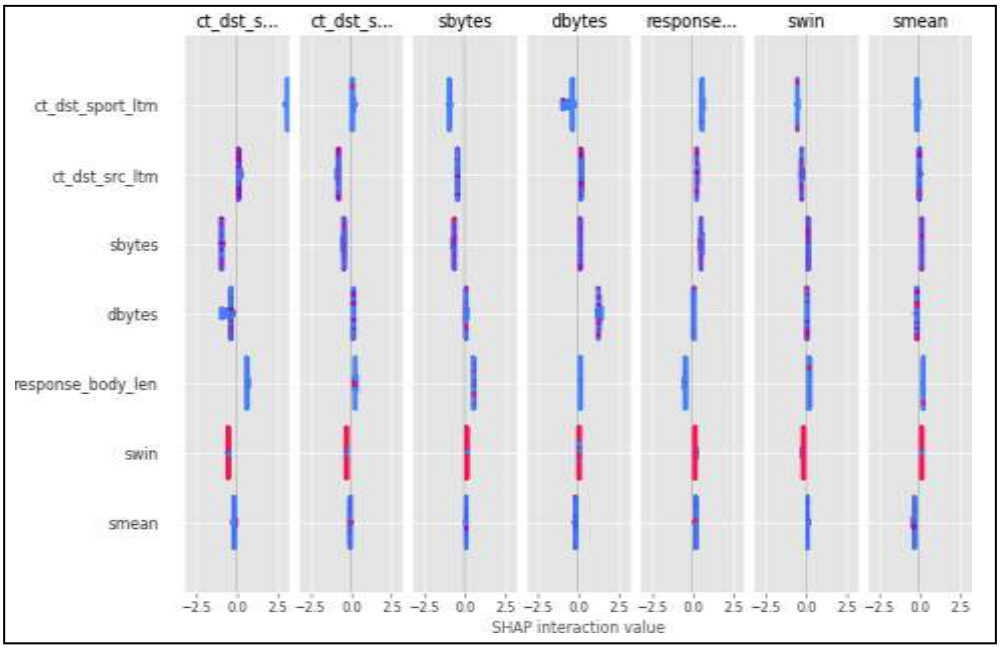


Figure 14: SHAP Interaction Value Summary Plot

Furthermore, using the LIME package as used for the MLP Classifier, individual predictions of the XGBoost Classifier

can be explained.

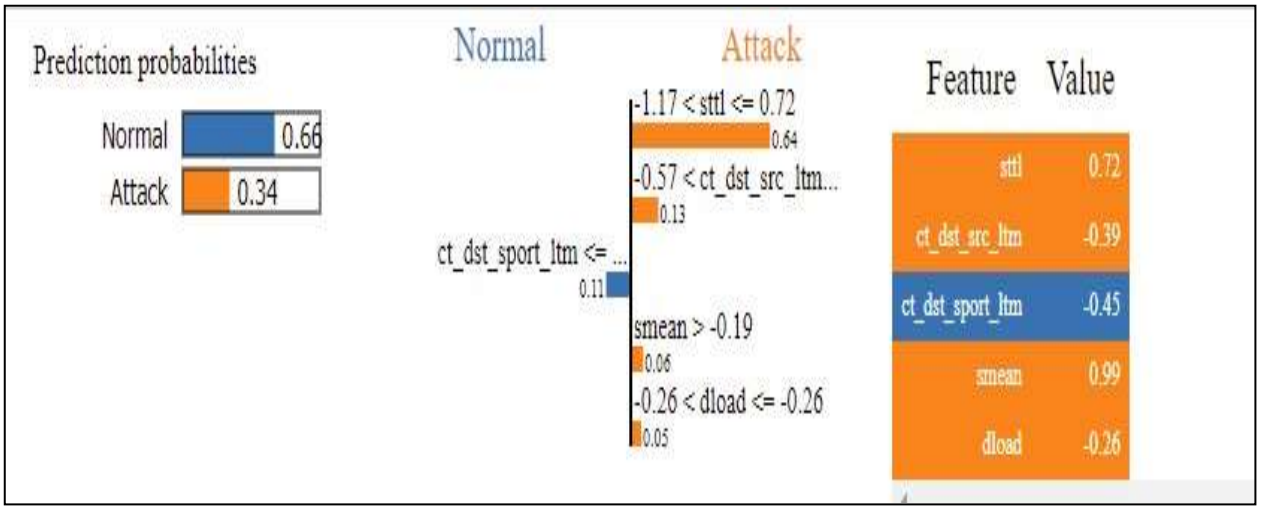


Figure 15: Single Classification Prediction using the XGBoost Classifier Explanation

The model offers highly efficient and flexible, while a high-performing classifier that can be paired with the SHAP and LIME libraries offers robust explainability features. This will increase the trustworthiness of advanced black-box algorithms for effectiveevaluations of ML-based IDSs for IoT network security.

5. Result Analysis

Performance metric

Performance metrics accurately select the important feature of a model and the parameters or model formulations in the model are adjusted to minimize errors and the appropriate model is selected. Therefore, this study has adopted the classification performance measurement method to properly analyze and classify the features in the dataset required to identify intrusions in the IoT network. The classification performance measurement approach uses metrics called true positive that show the ratio of the number of time steps properly classified as "attack detection".

For the performance, the measurement used four parameters True Positives (TP), False Positives (FP), True Negatives (TN), and False Negatives (FN)

$$Precision = \frac{TP}{TP + FP}$$

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

$$F1 - Score = 2 \times \frac{recision \times Recall}{Precision + Recall}$$

$$Acc. = \frac{TP + TN}{TP + TN + FP + FN}$$

Comparative Result AnalysisCross-Validation

Evaluating the performance of the model, we used the k-fold cross-validation technique and find out which classifier gives the best performance for feature selection in intrusion detection over the UNSW-NB15 dataset. In the cross-validation, we measure the AUCand cross-validation time. Table 1 shows the analysis of Cross-Validation with various Machine learning models for feature selection. It found that the logistic regression classifier gives the best performance it gives the 96.45% AUC test and the cross- validation time is 3.65 sec.

Step of k-fold Techniques

Step 1: Set k = 5

Step 2: Split the dataset into k-fold

Step 3: Select the k-1 folds for the training set, and the remaining for the test setStep 4: Train the models as the training data set

Step 5: Validate a test data set

Step 6: Stored the final results of the validationStep 7: Repeat steps 3-6 k-times

Step 8: To get the final average score on step 6.

Table 1: Analysis of Cross-Validation with various Machine learning models for feature Selection

Model Name	CV Fit Time	CV Accuracymeans	CV Precisionmean	CV Recall mean	CV F1 mean	CV AUC mean	Test Accuracy	Test Precision	Test Recall	Test F1	Test AUC
Random Forest	13.658747	0.977093	0.983078	0.975183	0.979114	0.997192	0.898301	0.987719	0.861288	0.920181	0.981120
Decision Tree	1.176954	0.965773	0.968616	0.969249	0.968930	0.965390	0.890299	0.979849	0.856437	0.913995	0.909480
Multi-Layer Perceptron	195.017369	0.961789	0.966588	0.963999	0.965251	0.994191	0.887568	0.985148	0.847588	0.911205	0.979146
Logistic Regression	3.651156	0.879500	0.877955	0.907284	0.892374	0.959014	0.873424	0.959241	0.850152	0.901408	0.964595

Table 2: Accuracy score of Feature Selection of Explainable AI with machine learning classifiers

Model Name	Accuracy
Explainable AI (XAI) with a Decision Tree Classifier	0.85208
Explainable AI (XAI) with an MLP (Multi-Layer Perceptron)	0.97915
Explainable AI (XAI) with an Xgboost	0.89829

Table 2 shows the accuracy score of Feature Selection of Explainable AI with machine learning classifiers. It is found that after integrating the proposed Explainable AI classifier with the machine learning techniques, Multilayer perceptron gives a better accuracy score which is 97.15%. Table 2 shows the accuracy score of the feature selection of explainable AI with machine learningclassifiers. It has been found that when combined proposed explanatory AI classifier with the machine learning technique, the multilayer perceptron gives a higher accuracy score which is 97.15%. In this study proposed explanatory AI was not combined withthe logistic regression technique because after cross-validation logistic regression gave better results which are 96.45.

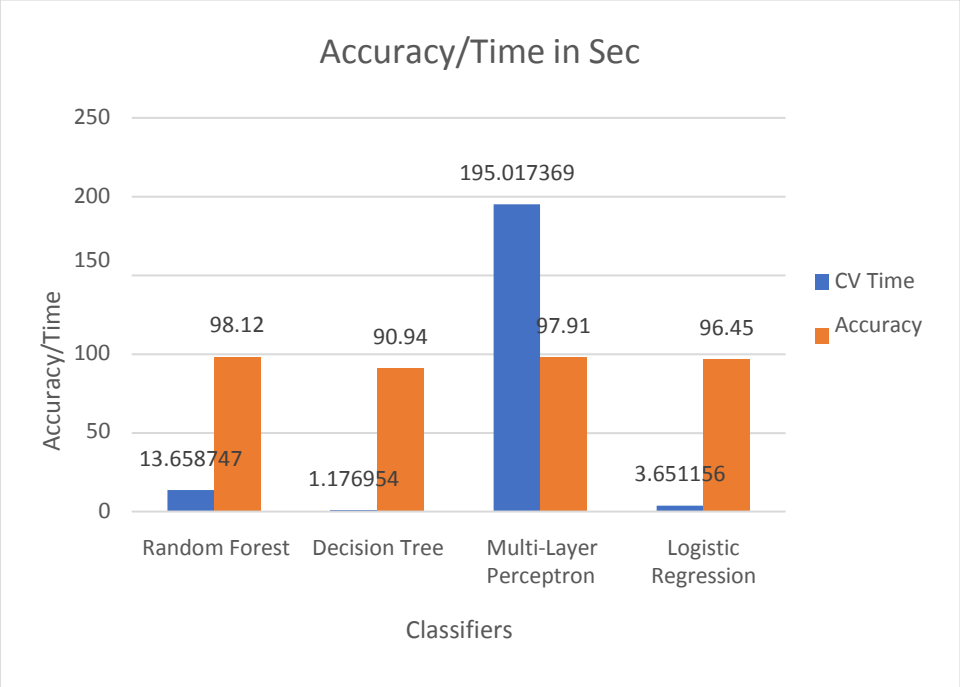


Figure 16: Comparative Analysis of Various Classifiers with the Accuracy score and Time

Figure 15 shows the comparative analysis of various classifiers with the accuracy score and time of classification for optimal featureselection.

6. Conclusion

ML learning models utilized for IoT network traffic security through IDSs are increasingly becoming more complex, but the need for human analysts to analyze outcomes through inherent domain knowledge for resource allocation and cyber security strategy development is a critical role. ML algorithms are often considered "black boxes", in which the logic or explanation behind the output predictions is not interpretable. By utilizing the UNSW-NB15 dataset and training a Decision Tree, MLP, XGBoost, and logistic regression. The accuracy of the feature selection shows high performance for analyzing network behavior of Attack or Normal Activity between connected clients in an IoT network. After analyzing the performance of ML classifiers, established libraries and techniques for enabling explainability or Explainable AI (XAI) were applied to the trained classifiers to explain their decisions and evaluate the important features. In the immediate term, this increased transparency will increase trust with ML systems in the IoT cyber security domain. Ultimately, it will enable a new range of capabilities of IoT cyber security through extracting insights from sophisticated machine learning models as more explainability conveys the influence of the influence the prediction of a cyber-attack and to what degree. In [25] used methods for result visualization.

We discovered what collection of feature values might identify malicious from typical network traffic by examining the important features based on the Explainable AI classifier and SHAP technique. This paper evaluates the performance of the model based on the k-fold cross-validation technique and finds which classifier gives the best performance for feature selection in intrusion detectionover the UNSW-NB15 dataset. In the cross-validation, Test the AUC and cross-validation time. It found that the logistic regression classifier gives the best performance it gives the 96.45% AUC test and the cross-validation time is 3.65 sec. In this study proposed explanatory AI was not combined with the logistic regression

technique because after cross-validation logistic regression gave betterresults which are 96.45.

References

[1] D. Pienta, S. Tams, and J. atcher, "Can trust be trusted in cyber security?" in Proceedings of the 53rd Hawaii InternationalConference on System Sciences, Maui, HI, USA, January 2020.

[2] Tankard, C. Big data security. Netw. Secure. 2012, 2012, 5-8.

[3] Khan, M.; Karim, R.; Kim, Y. A Scalable and Hybrid Intrusion Detection System Based on the Convolutional-LSTM Network.Symmetry 2019, 11, 583.

[4] Meryem, A.; EL Ouahidi, B. Hybrid intrusion detection system using machine learning. Netw. Secur. 2020, 2020, 8–19.

[5] Sarker, I. H.; Kayes, A.S.M.; Badsha, S. Alqahtani, H.; Watters, P.; Ng, A. Cyber security data science: An overview from amachine learning perspective. J. Big Data 2020, 7, 1-29

[6] A. B. Arrieta, N. Diaz-Rodriguez, J. Del Ser, et al., "Explainable artificial intelligence (XAI): Concepts, Taxonomies,Opportunities, and Challenges Toward Responsible AI," Information Fusion, vol. 58, pp. 82-115, 2020.

[7] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep inside convolutional networks: Visualising image classification models andsaliency maps," arXiv preprint arXiv:1312.6034, 2013.

[8] R. Iyer, Y. Li, H. Li, M. Lewis, R. Sundar, and K. Sycara, "Transparency and explanation in deep reinforcement learning neuralnetworks," in Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society. ACM, 2018, pp. 144-150.

[9] S. Mishra, B. L. Sturm, and S. Dixon, "Local interpretable model-agnostic explanations for music content analysis." in ISMIR,2017, pp. 537-543.

[10] S. Anjomshoae, K. Främling, and A. Najjar, "Explanations of black-box model predictions by contextual importance and utility," in International Workshop on Explainable, Transparent Autonomous Agents and Multi-Agent Systems. Springer, 2019, pp.95-109.

[11] S. M. Othman, F. M. Ba-Alwi, N. T. Alsohybe, and A. Y. Al- Hashida, "Intrusion detection model using machine learning algorithm on big data environment," Journal of Big Data, vol. 5, no. 1, p. 34, 2018.

[12] K.V.V.N.L. Sai Kiran, R.N. Kamakshi Devisetty, N. Pavan Kalyan, K. Mukundini, R. Karthi, (2020) Building an Intrusion Detection System for IoT Environment using Machine Learning Techniques, Procedia Computer Science, Volume 171, Pages 2372-2379, https://doi.org/10.1016/j.procs.2020.04.257.

[13] Albulayhi, K., Smadi, A. A., Sheldon, F. T., & Abercrombie, R. K. (2021). IoT Intrusion Detection Taxonomy, Reference Architecture, and Analyses. Sensors, 21(19), 6432. https://doi.org/10.3390/s21196432.

[14] S. V. Farrahi and M. Ahmadzadeh, "KCMC: a hybrid learning approach for network intrusion detection using k-means clustering and multiple classifiers," International Journal of Computer Applications, vol. 124, no. 9, 2015.

[15] S. Paliwal and R. Gupta, "Denial-of-service, probing & remote to user (R2L) attack detection using genetic algorithm," International Journal of Computer Applications, vol. 60, no. 19, pp. 57–62, 2012.

[16] K. Peng, V. Leung, L. Zheng, S. Wang, C. Huang, and T. Lin, "Intrusion detection system based on decision tree over big datain fog environment," Wireless Communications and Mobile Computing, vol. 2018, Article ID 4680867, 10 pages, 2018.

[17] L. H. Gilpin, D. Bau, B. Z. Yuan, A. Bajwa, M. Specter, and L. Kagal, "Explaining explanations: an overview of Interpret ability of machine learning," in Proceedings of 2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA),pp. 80–89, IEEE, Turin, Italy, October 2018.

[18] P. Svenmarck, L. Luotsinen, M. Nilsson, and J. Schubert, "Possibilities and challenges for artificial intelligence in military applications," in Proceedings of the NATO Big Data and Artificial Intelligence for Military Decision-Making Specialists' Meeting,Bordeaux, France, May 2018.

[19] M. Stampar and K. Fertalj, "Artificial intelligence in network intrusion detection," in Proceedings of the 2015 38th InternationalConvention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), pp. 1318-1323, IEEE, Opatija, Croatia, May 2015.

[20] W. Lee, S. J. Stolfo, P. K. Chan, et al., "Real-time data mining-based intrusion detection," in Proceedings of the DARPA Information Survivability Conference and Exposition II. DISCEX'01, pp. 89-100, IEEE, Anaheim, CA, USA, June 2001.

[21] Moustafa, N.; Slay, J. UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). In Proceedings of the 2015 Military Communications and Information Systems Conference (MilCIS), Canberra, Australia,10-12 November 2015.

[22] Moustafa, Nour & Slay, Jill. (2015). UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set). 10.1109/MilCIS.2015.7348942.

[23] T. Zahavy, N. Ben-Zrihem, and S. Mannor, "Graying the black box: understanding DQNs," in Proceedings of the InternationalConference on Machine Learning, pp. 1899-1908, New York, NY, USA, 2016.

[24] M. Wang, K. Zheng, Y. Yang and X. Wang, "An Explainable Machine Learning Framework for Intrusion Detection Systems,"in IEEE Access, vol. 8, pp. 73127-73141, 2020, doi: 10.1109/ACCESS.2020.2988359.

[25] S. M. Lundberg, B. Nair, M. S. Vavilala, M. Horibe, M. J. Eisses, T. Adams, D. E. Liston, D. K.-W. Low, S.-F. Newman, J. Kim et al., "Explainable machine-learning predictions for the prevention of hypoxaemia during surgery," Nature biomedical engineering, vol. 2, no. 10, p. 749, 2018