International Journal of Computing and Digital Systems
2025, VOL. 17, NO. 1, 1–13

http://dx.doi.org/10.12785/ijcds/1571030973

Demystifying IoT Network Intrusion Detection : Assessing ML Algorithms with the Aid of Explainable AI

Tasfia Zaima[1], Tabassum Ibnat Ena[1], Md. Tamim Ikbal[1] and Abu Sayed Md. Mostafizur Rahaman[2]

[1]*Department of Information and Communication Technology, Bangladesh University of Professionals, Dhaka, Bangladesh*
[2]*Department of Computer Science and Engineering, Jahangirnagar University, Dhaka, Bangladesh*

Received 16 May 2024, Revised 10 January 2025, Accepted 11 January 2025

Abstract: Intrusion Detection Systems (IDSs) are pivotal for network security; while machine learning based IDSs surpass traditional models in effectiveness, their growing complexity poses transparency challenges. This study uses the UNSW-NB15 dataset to train the ML algorithms, aiming to demystify the complexity of IoT network intrusion detection. The Explainable Artificial Intelligence (XAI) framework is used to improve model comprehensibility and transparency. Scikit-Learn, ELI5 Permutation Importance, and Local Interpretable Model-Agnostic Explanation (LIME) are applied to analyze the performance of many ML algorithms. This study also investigates the influence of dataset balancing on the performance metrics of various ML algorithms. SVM accuracy rose from 86 to 88 percent, while Random Forest and CatBoost accuracy climbed from 90 to 92 percent after balancing. Ensemble combinations also showed improved performance. ELI5 and LIME were then applied to the ML algorithms. The methodology presented in this paper offers a valuable toolkit for cybersecurity experts, empowering them to make informed decisions in the face of evolving cyber threats. The findings support the integration of XAI approaches with conventional ML systems to improve interpretability in cybersecurity applications. This study enhances IDSs for IoT networks by bridging the gap between ML-based prediction performance and the need for transparent and interpretable decision-making.

Keywords: Explainable AI (XAI), Random Forest (RF), SVM, CatBoost, ELI5, LIME, Permutation Importance (PI)

1. INTRODUCTION

IoT networks, vast reservoirs of user data, are becoming more and more integrated in our daily lives and are critical to the delivery of many crucial services. Our daily activities rely so largely on the internet that it has become a target for hackers. Intrusion detection systems (IDSs) are hardware or software solutions that automate the act of keeping an eye on events happening within a network or computer system and interpreting them to look for indicators of potential security issues. In the last several years, the frequency and intensity of network attacks have increased, making IDSs an essential component of every organization's security infrastructure. They accomplish this by employing a range of ML algorithms that, if any suspicious activity or possible cyberattack is detected, alerts are produced [1].Some of the basic drawbacks of the conventional IDS are presented by the system, including low power consumption when in use, absence of well-established IoT protocols and architecture, and limited computing capability. These systems are good at differentiating between attack and normal behaviors and in identifying apprehensive behaviors and activities. However, their "black box" design makes it strenuous to completely comprehend the decision-making processes that underlie within the predictions and attack classifications which causes lack of transparency and trust and cyber analysts find it hard to come up with feasible response to threats. Model explanation is essential to make optimal cybersecurity assessments and create powerful information assurance plans to countermeasure potential vulnerabilities by validating outputs given by IDSs. One of the effective ways to solve the issue of incomprehensability of black-box models is to incorporate Explainable AI (XAI) [2] which facilitates the analysts to comprehend the rationale behind the algorithm's decision. This study portrays an extensive overview of ML algorithms, with a particular emphasis on Random Forest (RF), Support Vector Machines (SVM), and CatBoost, and assesses how well they work in the context of IoT network security. The research uses XAI methodologies by using various combinations of ensemble techniques for these classifiers and by utilizing Python tools that enhance the explainability and transparency of complex ML models. Explain Like I'm 5 (ELI5) and Local Interpretable Model-Agnostic Explanations (LIME) are used to visualize feature importance, boosting the models' explainability in

E-mail address: tasfia.zaima000@gmail.com, tabassumibnat245@gmail.com, tamimmd12348@gmail.com, asmmr@juniv.edu

classification prediction scenarios. LIME and ELI5 include feature ranking that makes it easier to assess the in-depth effectiveness and efficiency of the selected machine learning algorithms. The problem at hand involves conducting a comprehensive evaluation of the real-world applicability of an IDS enhanced with XAI [3]. The specific objectives of this research are-

- To evaluate the real-world applicability of XAI enhanced IDS.
- To apply XAI to modify the ML algorithms to increase transparency and provide an explanation for the algorithmic choices.
- Using XAI to rank the features according to their importance.
- To assess ML algorithm performance with the aid of XAI.

The focus is given on the interpretability of ML algorithms applying XAI using the dataset called USNW-NB15.This dataset comprises of traffic-pattern based security issues of real-life scenerios of IoT devices.Here's what the research aimed to contribute:

- To improve trust management that is comprehensible to human professionals, the XAI concept was tackled. To do so, feature importance and some ML models were used.
- UNSW-NB15 dataset was balanced, and the performance metrics were compared to that of the imbalanced dataset. Some performance metrics improved because of making the dataset balanced improving accuracy and reliability.
- The features extracted from the models using XAI for IDS to enhance human interpretability were interpreted for better explainability.

This paper [4] shows the integration of ML and DL with XAI for implementation of IDS with AI. Naïve Bayes, KNN, DNN,RBF, SVM, XGBoost, RF, DT are used for analysis using the IEC 60870-5-104 Intrusion Detection Dataset and CIC-IoT-Dataset-2022 . For model explainability using XAI, SHAP generated feature importance that is helpful for security analysts to establish transparency. In this paper [5] possible challenges of IDSs by overviewing current ML and DL methods integrated with XAI techniques are outlined. The significance of XAI for global and local interpretation of different extracted features from different public datasets are reviewed here to improve the explainability of models. The focus of the study in [6] is a framework called FAIXID that improves the understandability of intrusion detection alerts by leveraging XAI and data cleaning techniques. Framework evaluation is done to understand whether the framework can fulfill the goal or not or if data cleaning can increase the explainability of IDS results or not. The explainability increases from 0.066 to 0.2056 due to data cleaning and there is scope to conduct comparisons here involving different XAI toolkits. The study presented in [7] suggests a hybrid IDS where close attention is paid to interpretability and explainability to populate the knowledge base with ML-suggested rules and maintain an environment that is understandable and interpretable for the IDS. The ML model was integrated and deployed using DT with Scikit-Learn to keep up justifying the dynamic system of IDS. There is scope to make the model described here more robust. This paper [8] created self-organizing maps (SOM) grounded on the X-IDS system, which generates visualizations that are explainable. SOM basically works to transform one-dimensional data to higher dimensions using this technique in the IDS to get both local and global explanations. Producing reliable IDS and visualization such as feature importance, U-matrices and feature heatmaps were the primary goals. 91 percent and 80 percent accuracy in the datasets-NSL-KDD and CIC-IDS-2017 respectively were achieved. There is scope to improve the model to get higher accuracy and work with all kinds of malicious attacks instead of DDoS attacks only. This paper [9] shows experimental analysis to enhance AI-based In-Vehicle (IV-IDS) IDS as the primary goal. It mainly tackles the problem of false alarms, where it presents a visualization-based explanation strategy called "VisExp" that, when given to experts, greatly enhanced their level of faith in the system in comparison to explanations based on rules. Basically, interpretation of anomaly detection applying XAI for cybersecurity in automobiles is explored. In the study [10] XAI integration with IoT is explored where cutting edge and future prospects are discussed. It highlighted the necessity of openness in AI choices, especially for IoT applications difficulties and potential prospects. This paper [11] proposed an IDS using two ML algorithms, where occurred implementation of classifiers DT and SVM. In KDD-Cup dataset, Particle Swarm Optimization (PSO) was applied and then the hybrid approach was implemented to lessen the FPR to 0.9 percent and achieve an accuracy of 99.6 percent. The focus of this paper [12] is on the overview of different approaches to improve interpretability of IDS to make more effective decisions regarding different cyberattacks handled by CSoC analysts. The lack of explanation of white-box and black-box models are explored here, and a three-tier model is proposed for X-IDS which is not restricted to any IDS solution. Seven black-box models were implemented in three datasets namely- RoEduNet-SIMARGL2021, NSL-KDD, and CICIDS-2017 in [13] to assess the performance. Then XAI was applied to them to emphasize the importance of feature selection through XAI which gives better explanation than without XAI in previously conducted comparisons. The NSL-KDD dataset is trained with XAI to produce interpretable outputs to make it more understandable to cybersecurity experts in [14] which enables them to make wiser decisions regarding any kind of cybersecurity breaches. It ultimately improves the confidence of cybersecurity analysts in taking any

decision regarding cyber-attacks detected through IDS. This paper [15] contributes to helping security experts in better understanding of an IDS by optimizing the structure based on explanation from classifiers which may be one-vs-all classifier or multiclass classifier. The proposed framework provides different interpretations using SHAP to provide better understanding of judgements pertaining to the IDS.

In modern time, with advancements in technology and widespread use of the internet, information security has become critically important. XAI techniques in this work are layered upon the ML algorithms and in particular LIME. Here, importance of balancing of the dataset is emphasized as this improves the performance and accuracy of the models as well as the resilience and dependability of the model.

In Section 2, a detailed explanation of the methodology is provided. The proposed methodology and an explanation of the XAI techniques can be observed from there. The result analysis is presented in Section 3. The paper is concluded in Section 4.

2. Methodology

An outline of the proposed IDS model architecture is shown in Figure 3. Here, the model's steps are displayed one after the other. The pre-processing steps include data cleaning, normalization, balancing, and transformation which are displayed here for the UNSW-NB15 dataset. Then in the segment of model training, the classifiers Random Forest, SVM, CatBoost are trained using the UNSW-NB15 dataset. Besides these 3 supervised classifiers, ensemble learning method is applied. The performance of ensemble approaches using stacking and voting classifiers are also observed. Basically, a performance comparison is obtained from the classification report achieved by training these classifiers. After doing so, ELI5 and LIME, which are the XAI approaches to the interpretability of the model are applied to get a better understanding and interpretation of the model. Feature importance which is defined as scores used to determine the relative extent of individual features in a dataset while constructing a predictive model is obtained.Then model explanation is a collection of procedures and techniques that, as opposed to relying just on blind faith, enable users to comprehend and value the output of a ML algorithm. By using XAI, it is possible to explain the model, which will enhance the reliability and interpretability of the end users. The dataset collection, data processing steps, classifiers utilized for analysis, and proposed methodology covering XAI techniques ELI5 and LIME are the main subjects of Section 2.

A. Dataset

The UNSW-NB15 computer network security dataset [16] is a product of the Cyber Range Lab at the University of New South Wales Canberra, which was made available in 2015, served as the source material for this study. The dataset contains 2,540,044 examples of both normal and abnormal network behavior of IoT devices, produced through the IXIA PerfectStorm program. The dataset consists of three sets of attributes: fundamental, content, and time. In Figure 3, the intrusion detection model proposed is illustrated.

B. Pre-Processing

Data pre-processing is an essential process for organizing data that is structured as well as unstructured for any kind of data analysis or data mining. It basically converts raw data into suitable formats for machine learning. Applying feature selection and associated techniques can enhance the quality of features within a dataset and the insights obtained. The two primary categories are numerical and categorical features. Categorical Features are features that describe variables with fixed and limited values and Numerical Features are the continuous features that can take on a range of numerical values are known as numerical features [17].

Machines like to work with ordered and organized data like processed texts, photos, and videos. If these are unprocessed it becomes hard for machines to work with them. So, using data pre-processing techniques these data need to be cleaned and processed for usage.

1) Data Cleaning

Data cleaning involves removing erroneous, duplicate, or incomplete information from databases and is an essential stage in data management. It is a component of the pre-processing step, where the objective is to fix inaccurate data, eliminate redundant information, and deal with missing or incomplete information.

2) Normalization

To facilitate faster querying and analysis and improve corporate decision-making, data normalization is a procedure that helps establish a uniform data format throughout a system. To provide a more logical storage strategy, it entails rearranging data sets to eliminate unstructured or unnecessary information. To scale values to a common range so they can be compared to other data sets, data normalization formulas are utilized. The formula modifies the variation of the data set between 0 and 1, where the lowest data point has zero and the highest has a one-valued normalized value. The other data points have decimal values ranging from zero to one. Mathematically, the normalizing equation looks like this [18] :

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)} \tag{1}$$

3) Balancing

The UNSW-NB15 dataset was used in the study which was a case of imbalanced dataset at first. Random undersampling is used here to make the UNSW-NB15 dataset balanced in this pre-processing step of dataset analysis. In Figure 1, it is seen that the class distribution of label 0 and label 1 are not balanced. Before undersampling the

class distribution be as Figure 1.Balancing is done through the process of undersampling which means that the values from class label 1 were discarded to balance the dataset. In case of imbalanced, the support count.for class 0 was 56000 and for class 1 was 119341. After undersampling, the support count for both class 0 and class 1 became 56000. The different categories of attack also had different counts and distribution which were balanced too.

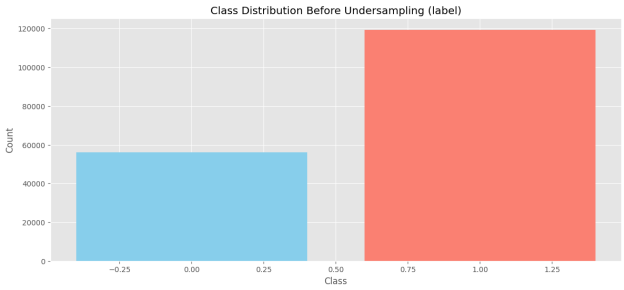

Figure 1. Imbalanced Dataset Class Label

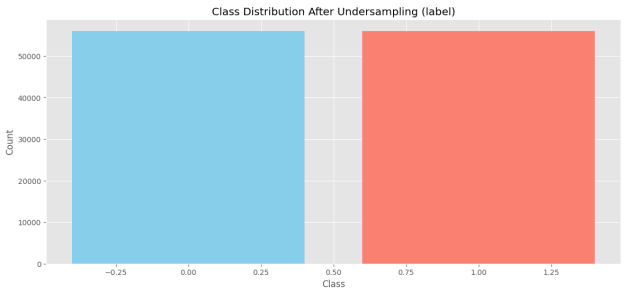

Figure 2. Balanced Dataset Class Label

After doing random undersampling of the majority class, which is label 1, we obtain balanced distribution of class label in Figure 2.

4) Transformation

The act of transforming, cleaning, and organizing data into a format that can be used for analysis and assist decision-making processes to advance an organization's expansion is known as data transformation.Data munging and data wrangling are other terms for data manipulation. Data transformation will begin the process of transforming the data into the format needed for analysis and subsequent processes, which we have already begun with data cleansing [19].

C. Model Implementation

To do implementation of the IDS model as Figure 3, the ML algorithms as follows are to be taken and trained for implementation.

1) Random Forest

Random Forest is an ensemble method to manage supervised categorization. Random forests stem from building decision trees to using training sets and supervised learning techniques to improve the accuracy of the algorithms. The bagging technique is used by Random Forest to construct decision tree ensembles. Random forests generate several decision trees based on random data selections. The primary benefit of random forests is that they make less classification errors [20].

2) Support Vector Machine

SVM algorithms purpose is to locate a hyperplane that can discriminate between data points, where N is the number of attributes, to minimize computational time when dealing with millions of samples. This approach lowers the classification risk rather than attempting to achieve the best classification. Utilizing hyperplanes, data points that fall into several groups according to their location on the hyperplane can be categorized. Furthermore, the hyperplane's dimensions are determined by the number of features. A line can only be a hyperplane if it has two input features. The data points that establish a hyperplane's orientation and position are called support vectors. The SVM is the most dependable and effective model and classification technique in the high dimensional feature space for two ad hoc classification issues between two classes [21].

3) CatBoost

A potent machine learning method that has produced exceptional results in a variety of applications is the CatBoost algorithm. However, CatBoost was designed to deal with qualities that are categorical. It can still handle properties that are continuous or numerical. The gradient-boosting decision tree technique now incorporates the cat boost model as a special feature. There are GPU and CPU implementations for CatBoost. It is based on decision trees like other boosting approaches (e.g., XGBoost, LightGBM).The main notion of CatBoost is to sequentially add trees where each new tree tries to fix the faults committed by the previous ones. CatBoost includes various advances, like ordered boosting and an efficient handling of categorical variables, making it stand out in the family of boosting algorithms [22].

4) Ensemble Methods

The ensemble methods that were used in this work are voting and stacking. Voting is one way that combines several techniques to produce improved outcomes. Utilizing this approach, we must first create several categorization models utilizing the dataset for training. In our code the soft voting technique was utilized to achieve the results. Soft voting is used in classification of our research work.

In this approach, instead of each model in the ensemble voting for a single class (hard vote), they predict the probability of each class. The final output class is determined by averaging the probabilities given by all the models in the ensemble. On the other hand, Stacking, which is an ensemble ML technique is also analyzed. The center level classifiers in the stacking ensemble approach are learned

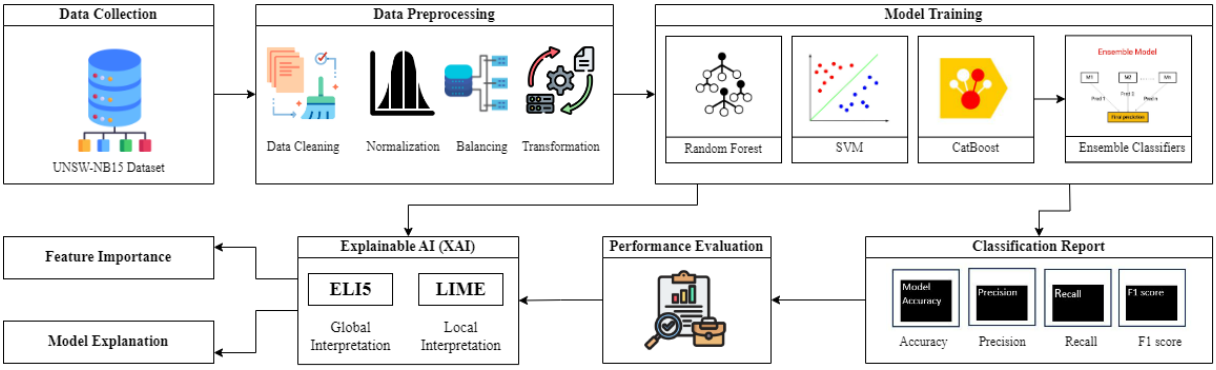

Figure 3. The Intrusion Detection Model Architecture

using the entire training dataset. The predictions made by the basic learners are sent into the Meta classifier as input and are handled as a fresh dataset [23].

D. Model Explanation using Explainable AI

In our model, we have used two popular techniques in the field of XAI which are ELI5 (Explain Like I'm 5) and LIME (Local Interpretable Model-agnostic Explanations). They contribute to the transparency and comprehensibility of complicated ML models by shedding light on their decision-making processes [24].

1) LIME

LIME is defined as a framework designed or library that provides explanations for machine learning models' predictions. LIME provides local interpretation, examines each of the model's individual predictions, and tries to explain the model's choices. The way LIME XAI operates is by producing justifications for each forecast.Firstly, LIME works with some perturbed samples and for each sample, it generates several instances creating new datasets which usually diverges from their initial instances causing a bit of change of points.The locally perturbed model can be estimated in the following way.

$$\xi(x) = \arg\min_{g \in \mathcal{G}} \{\mathcal{L}(f, g, w^x) + \Omega(g)\} \quad (2)$$

Here, g denotes the explanation model whose complexity is measured by $\Omega(g)$, f is the initial model, $\mathcal{L}$ is the loss, and w^x is the weight between sampled and initial data [15].

The new perturbed instances are then made to predict probabilities using black-box models to compare with the outcomes from previous samples before perturbation.In this way, new dataset created is trained using different ML models such as DT,inear Regression and logic regression and those give explanation of the predictions by generating the feature importance.The feature importance is shown through the help of visualization using LIME.The feature importance realized through LIME can positively of negatively impact the outcome.

2) ELI5

To elevate the interpretability of ML algorithms used, ELI5 is applied, which is robust and intuitive and is a Python library. Individuals can ask questions in the ELI5 subreddit and receive clear, concise answers in return. Regardless of experience level or background, the subreddit aims to make complex subjects understandable to all users. ELI5 provides explanations for each forecast, allowing users to understand the steps of a model.The contributions of the features are arranged based on weights after figuring out the priority weights to explain the importance of feature in achieving the final prediction.

In the last part of the model of Figure 3, we see the feature importance and model explanation through which we can do result analysis.

3. Result Analysis

The proposed models' behavior and performance were explained using XAI techniques that made use of ELI5 and LIME to facilitate performance improvement. The aim was to generate an IDS model that could explain the classification predictions and offer good accuracy. Using the UNSW-NB15 IoT-based network traffic dataset, the performance of the ML classifiers Random Forest, SVM, Catboost, and some ensemble stacking and voting algorithms was assessed through observing their classification reports in subsection A and B. Then, using XAI approaches, the normal and attack prediction probabilities in the ML classifiers were found and explained using ELI5 and LIME in subsection C and D.

A. Classification Report of Imbalanced and Balanced Dataset

The classification report is basically a performance evaluation metric of the ML algorithms that evaluates proposed model by classifying different instances into various classes. It basically evaluates how well the model works by measuring the trained models- Precision, F1 Score, Accuracy and Support. The classification reports containing the necessary information are illustrated here.

Classification Report for Random Forest Classifier:

	precision	recall	f1-score	support
0	0.78	0.96	0.86	56000
1	0.98	0.87	0.92	119341
accuracy			0.90	175341
macro avg	0.88	0.92	0.89	175341
weighted avg	0.92	0.90	0.91	175341

Figure 4. Classification Report of Random Forest (Imbalanced)

Classification Report for Random Forest Classifier:

	precision	recall	f1-score	support
0	0.88	0.96	0.92	56000
1	0.96	0.87	0.91	56000
accuracy			0.92	112000
macro avg	0.92	0.92	0.92	112000
weighted avg	0.92	0.92	0.92	112000

Figure 5. Classification Report of Random Forest (Balanced)

Classification Report:

	precision	recall	f1-score	support
0	0.72	0.93	0.81	56000
1	0.96	0.83	0.89	119341
accuracy			0.86	175341
macro avg	0.84	0.88	0.85	175341
weighted avg	0.89	0.86	0.87	175341

Figure 6. Classification Report of SVM (Imbalanced)

Classification Report for SVM Classifier:

	precision	recall	f1-score	support
0	0.85	0.93	0.89	56000
1	0.92	0.83	0.88	56000
accuracy			0.88	112000
macro avg	0.89	0.88	0.88	112000
weighted avg	0.89	0.88	0.88	112000

Figure 7. Classification Report of SVM (Balanced)

Classification Report:

	precision	recall	f1-score	support
0	0.77	0.98	0.86	56000
1	0.99	0.86	0.92	119341
accuracy			0.90	175341
macro avg	0.88	0.92	0.89	175341
weighted avg	0.92	0.90	0.90	175341

Figure 8. Classification Report of CatBoost(Imbalanced)

Classification Report for CatBoost Classifier:

	precision	recall	f1-score	support
0	0.87	0.98	0.92	56000
1	0.97	0.86	0.91	56000
accuracy			0.92	112000
macro avg	0.92	0.92	0.92	112000
weighted avg	0.92	0.92	0.92	112000

Figure 9. Classification Report of CatBoost(Balanced)

Classification Report:

	precision	recall	f1-score	support
0	0.76	0.97	0.85	56000
1	0.98	0.86	0.91	119341
accuracy			0.89	175341
macro avg	0.87	0.91	0.88	175341
weighted avg	0.91	0.89	0.89	175341

Figure 10. Classification Report of Stacking Classifier(Imbalanced)

Classification Report:

	precision	recall	f1-score	support
0	0.87	0.97	0.91	56000
1	0.96	0.85	0.90	56000
accuracy			0.91	112000
macro avg	0.91	0.91	0.91	112000
weighted avg	0.91	0.91	0.91	112000

Figure 11. Classification Report of Stacking Classifier(Balanced)

	precision	recall	f1-score	support
0	0.77	0.98	0.86	56000
1	0.99	0.86	0.92	119341
accuracy			0.90	175341
macro avg	0.88	0.92	0.89	175341
weighted avg	0.92	0.90	0.90	175341

Figure 12. Classification Report of Voting Classifier (Imbalanced)

Accuracy: 0.9197857142857143

	precision	recall	f1-score	support
0	0.87	0.98	0.92	56000
1	0.98	0.86	0.91	56000
accuracy			0.92	112000
macro avg	0.93	0.92	0.92	112000
weighted avg	0.93	0.92	0.92	112000

Figure 13. Classification Report of Voting Classifier(Balanced)

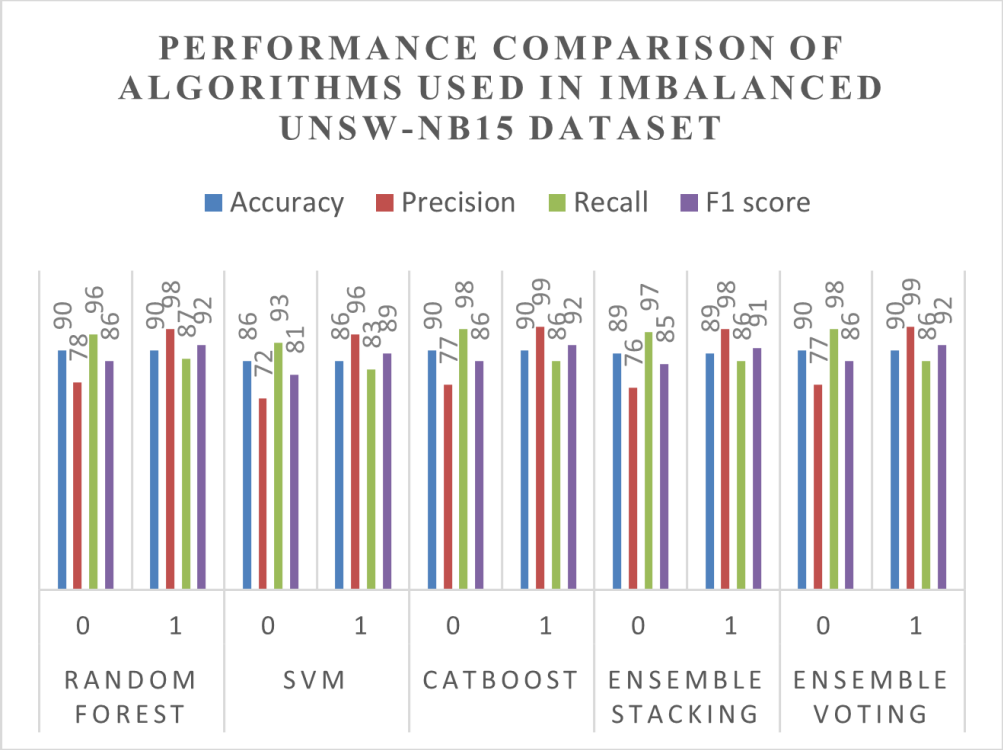

Figure 14. Performance Comparison of Algorithms Used in Imbalanced UNSW-NB15 Dataset

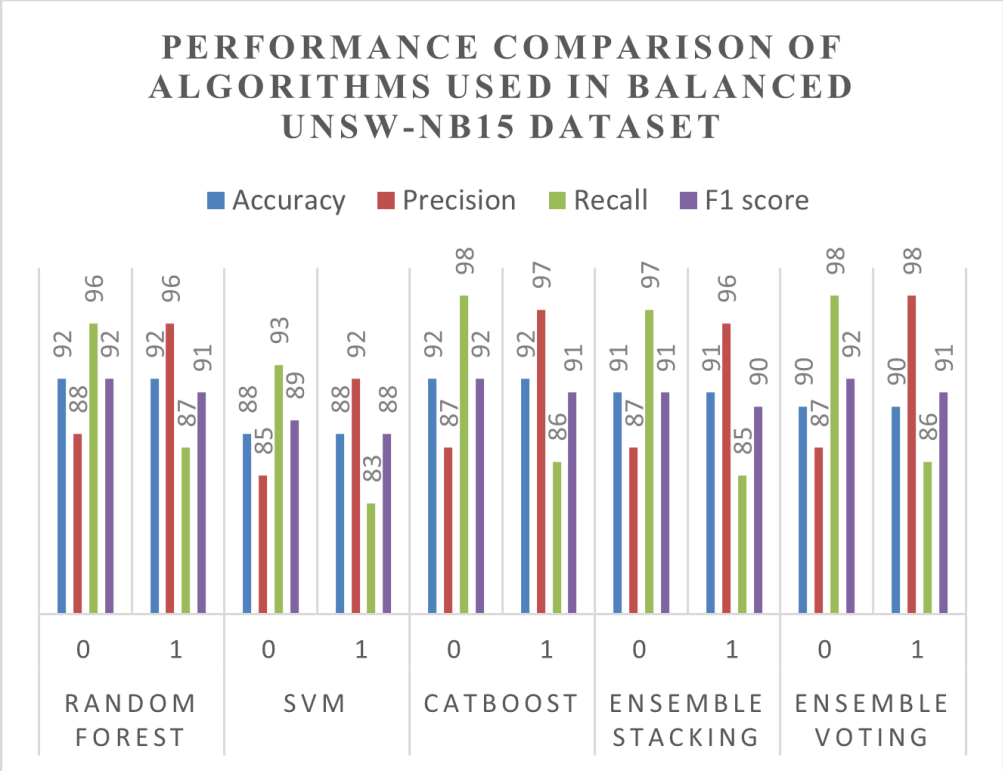

Figure 15. Performance Comparison of Algorithms Used in Balanced UNSW-NB15 Dataset

B. Performance Comparison

The comparison of performance of all ML algorithms in case of these algorithms being trained on the imbalanced and balanced UNSW-NB15 dataset is represented in Figure 14 and Figure 15. The performance evaluation measures - Recall, Accuracy, Precision, and F1 score for the classifiers that were used—RF, SVM, CatBoost, Ensemble Stacking and Ensemble Voting (Soft Voting) are presented in Figure 14 and Figure 15. These are shown for both the cases of imbalanced and balanced dataset. In the case of Figure 15, i.e. for balanced UNSW-NB15 dataset, 90 percent is the highest accuracy that comes amongst all classifiers. All these cases give accuracy of 90 or somewhat near that as observed in Figure 14 and Figure 15. Similarly, the maximum accuracy that the classifiers—RF and CatBoost—can provide for the imbalanced UNSW-NB15 dataset is 92 percent in Figure 14. Compared to Stacking (91 percent), Voting (90 percent) and other algorithms of classification, the ensemble method classifiers yield less accuracies. Besides, we see improvement in the precision values of class 0 and class 1 from imbalanced to balanced datasets. Accuracies and other metrics also improve in balanced dataset. When predicting different characteristics of the dataset, these performance metrics provide insightful information about how well-performing and reliable each algorithm is.

C. XAI ELI5 Implementation for Imbalanced and Balanced Dataset

The top features in the dataset, Permutation Importance (PI) or Mean Decrease Accuracy (MDA), which when run through any classifier yields some accuracy, were identified using the ELI5 PI toolkit and the Scikit Learn library. For classification prediction, improvement of explainability is essential to find the permutation importance module, which computes feature importance by observing how score drops when a feature is absent. And such is done for the Random Forest applied in the imbalanced and balanced UNSW-NB15 dataset to observe the weight of each feature. From Figure 18 and Figure 19, it can be observed that the features `ct_dst_src_ltm`, `ct_state_ttl`, and `sttl` are the top features. Amongst the top 3 features ,we find that two of the features matches. In the case of the other classifiers those are non-tree based. ELI5 is not applied to those other algorithms, rather we use LIME to find the interpretations. In the case of ML algorithms, it is one of the essential Python libraries that will aid in the model's interpretability and explainability. It helps in the overall decision-making process of any algorithm. On the basis of priority , the top weights are arranged from highest to lowest in the visualization through. As ELI5 is more appropriate for tree-based ML algorithms, ELI5 is performed for Random Forest for more accurate results. Observing Figure 18 and Figure 19, it is understandable that ELI5 is simpler than other XAI techniques.The important features are realized through applying ELI5 to find out the accuracy of predictions or outcomes performed by different ML algorithms.By identification of important features it is possible to remove bias and know if the model is taking right decisions or not.

Weight	Feature
0.0220 ± 0.0005	sttl
0.0171 ± 0.0005	ct_dst_src_ltm
0.0063 ± 0.0002	dttl
0.0052 ± 0.0004	ct_srv_dst
0.0026 ± 0.0001	ct_srv_src
0.0018 ± 0.0002	ct_state_ttl
0.0009 ± 0.0002	swin
0.0005 ± 0.0001	dmean
0.0004 ± 0.0001	spkts
0.0002 ± 0.0001	ct_src_ltm
0.0002 ± 0.0001	djit
0.0002 ± 0.0001	ct_dst_ltm
0.0000 ± 0.0000	is_sm_ips_ports
0.0000 ± 0.0001	ct_src_dport_ltm
0.0000 ± 0.0000	ct_flw_http_mthd
0 ± 0.0000	is_ftp_login
0 ± 0.0000	ct_ftp_cmd
0 ± 0.0000	trans_depth
-0.0000 ± 0.0000	response_body_len
-0.0001 ± 0.0001	sloss
… 19 more …	

Figure 18. ELI5 Permutation Importance for Random Forest (Imbalanced Dataset)

Weight	Feature
0.0220 ± 0.0004	sttl
0.0206 ± 0.0009	ct_dst_src_ltm
0.0048 ± 0.0003	ct_srv_dst
0.0041 ± 0.0002	dttl
0.0037 ± 0.0002	smean
0.0036 ± 0.0003	sload
0.0023 ± 0.0002	ct_srv_src
0.0020 ± 0.0005	dbytes
0.0012 ± 0.0001	ct_state_ttl
0.0011 ± 0.0004	sbytes
0.0004 ± 0.0001	ct_dst_ltm
0.0004 ± 0.0003	ct_dst_sport_ltm
0.0004 ± 0.0000	spkts
0.0003 ± 0.0000	ct_src_ltm
0.0002 ± 0.0001	ct_src_dport_ltm
0.0001 ± 0.0000	djit
0.0001 ± 0.0002	dmean
0.0001 ± 0.0002	swin
0.0000 ± 0.0000	is_sm_ips_ports
0 ± 0.0000	is_ftp_login
… 19 more …	

Figure 19. ELI5 Permutation Importance for Random Forest (Balanced Dataset)

D. XAI LIME Implementation for Imbalanced and Balanced Dataset

LIME is applied in the ML algorithms and the visual dashboard indicates the features and their relevant weights for which the classificatio is to be considered as 'Normal' or 'Attack' category in case of the imbalanced and balanced UNSW-NB15 network traffic dataset. This is useful to find various prediction probabilities and classification tasks of different ML algorithms.

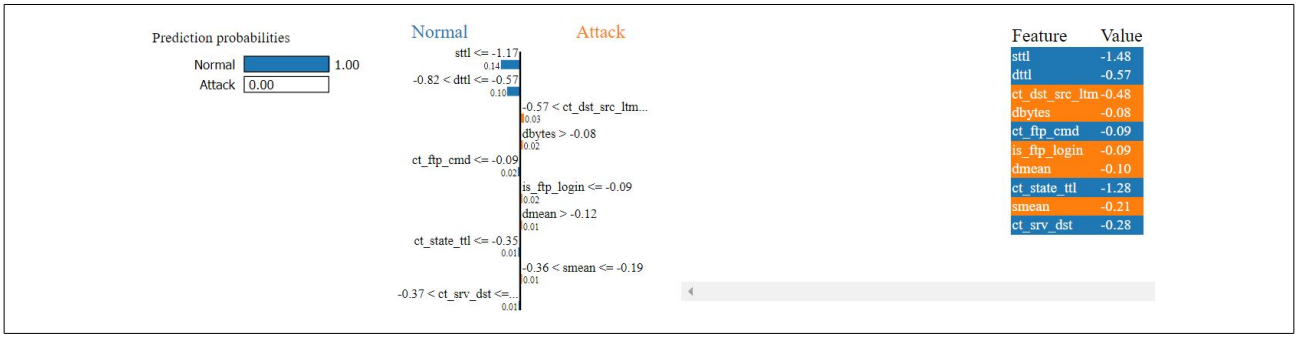

Figure 20. LIME Explanation of instance i =1653 of Random Forest in Imbalanced UNSW-NB15 Dataset

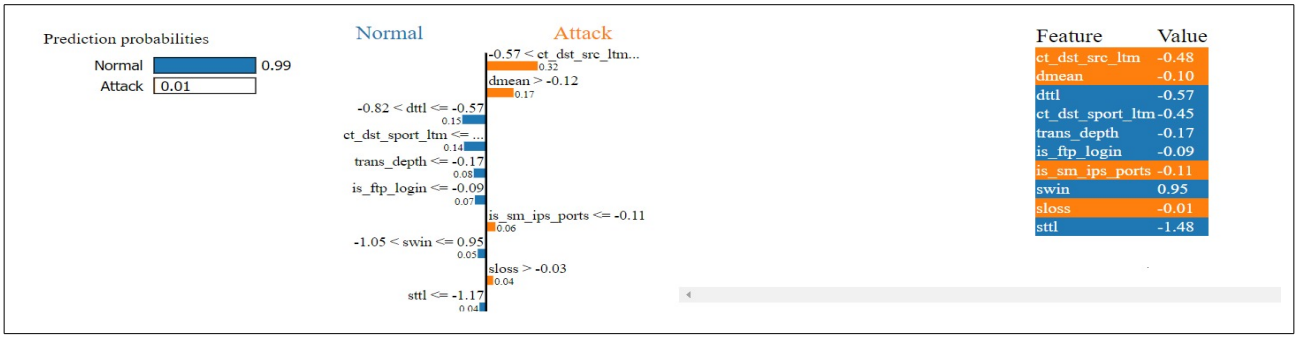

Figure 21. LIME Explanation of instance i =1653 of SVM in Imbalanced UNSW-NB15 Dataset

Figure 22. LIME Explanation of instance i =1653 of CatBoost in Imbalanced UNSW-NB15 Dataset

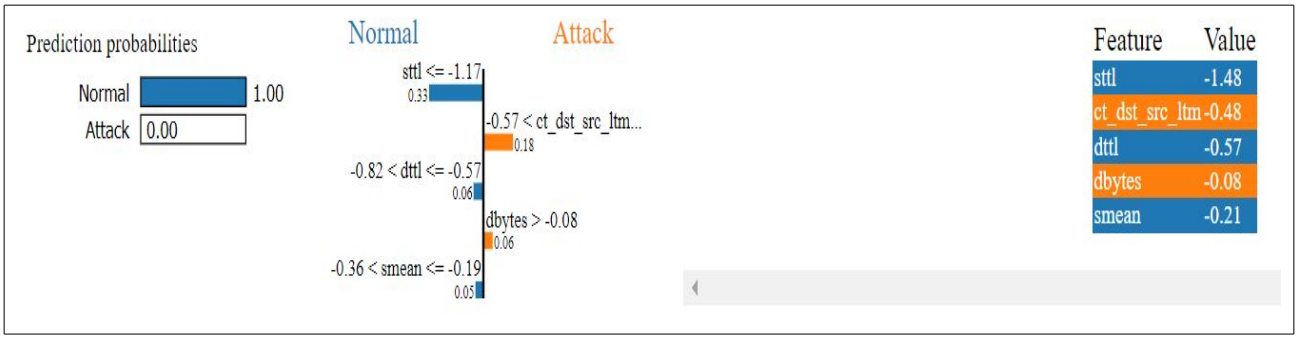

Figure 23. LIME Explanation of instance i =1653 of Stacking classifier in Imbalanced UNSW-NB15 Dataset

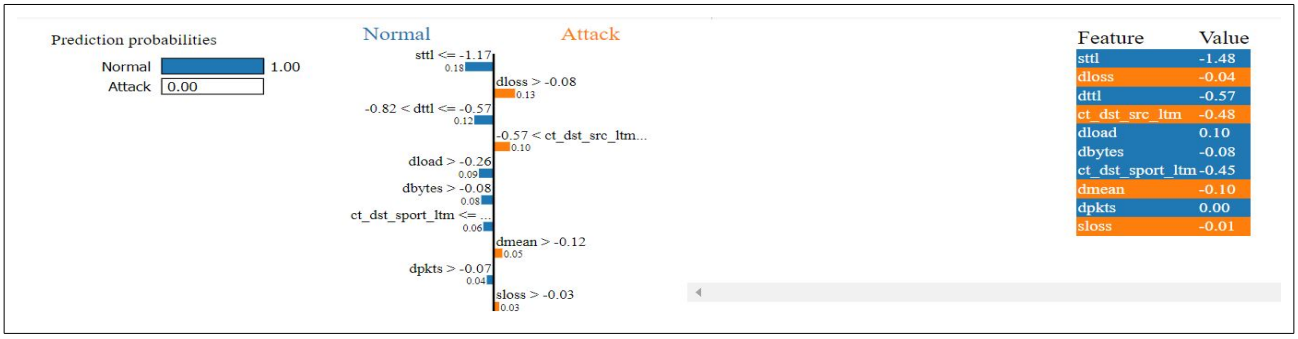

Figure 24. LIME Explanation of instance i =1653 of Voting classifier in Imbalanced UNSW-NB15 Dataset

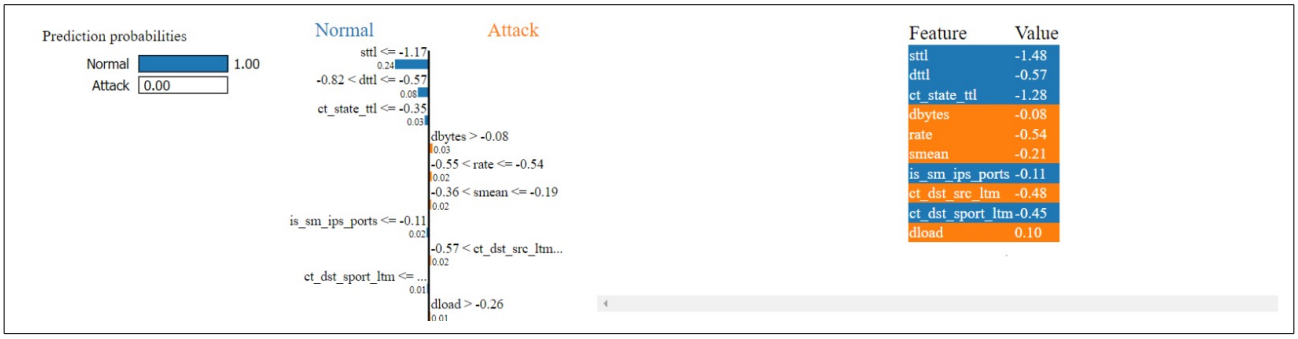

Figure 25. LIME Explanation of instance i =1653 of Random Forest in Balanced UNSW-NB15 Dataset

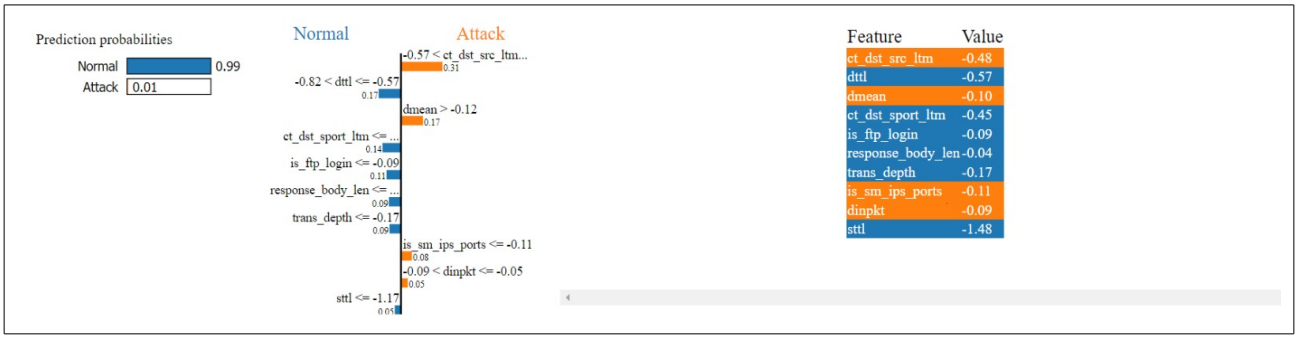

Figure 26. LIME Explanation of instance i =1653 of SVM in Balanced UNSW-NB15 Dataset

Figure 27. LIME Explanation of instance i =1653 of CatBoost in Balanced UNSW-NB15 Dataset

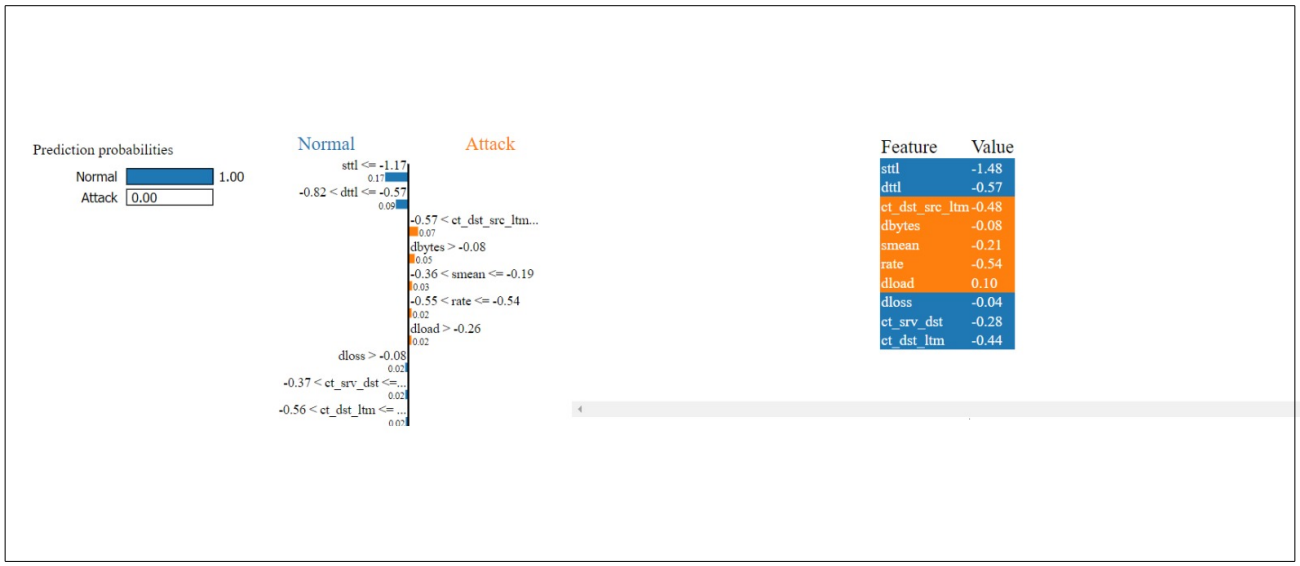

Figure 28. LIME Explanation of instance i =1653 of Stacking in Balanced UNSW-NB15 Dataset

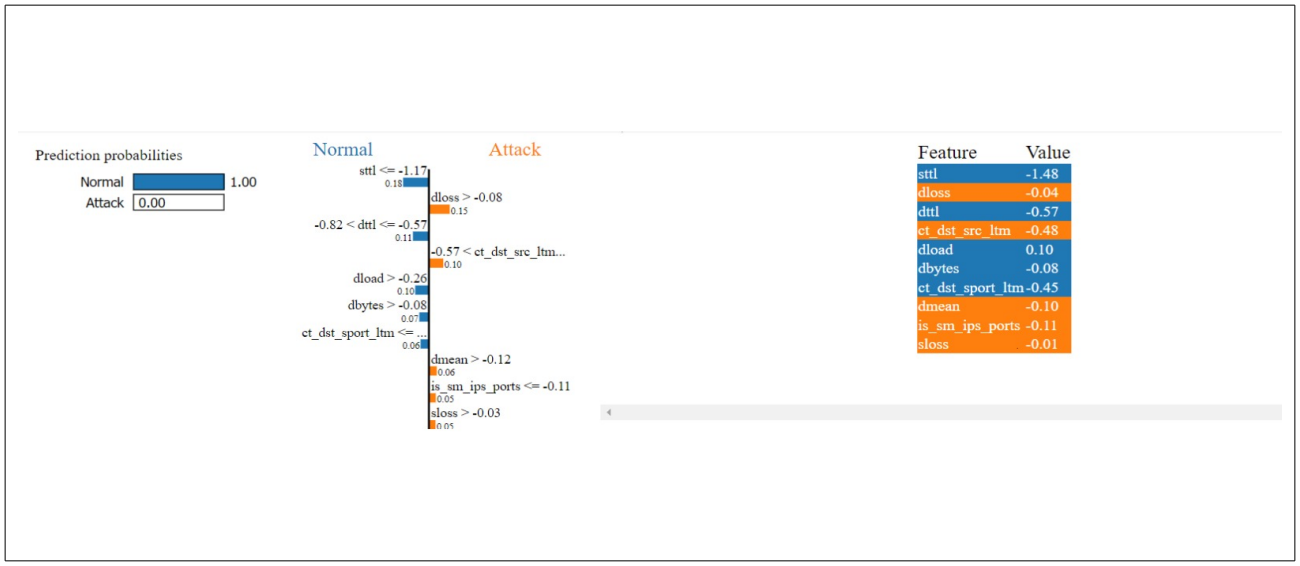

Figure 29. LIME Explanation of instance i =1653 of Voting in Balanced UNSW-NB15 Dataset

The visual dashboards obtained from LIME which are illustrated in Figure 20 to Figure 29 reveal which feature impact what category of prediction, and for which weight that prediction is obtained. The predictions may be 'Normal' or 'Attack' type. The features influencing 'Normal' prediction are denoted by blue and the features that influence the 'Attack' prediction are denoted by orange in the visual dashboard of it. The features and the values for which the prediction probabilities can be determined are also present in the visual dashboard of LIME which makes it easier to interpret and comprehend.

4. Conclusions and Future Work

To enhance explainability in the establishment of IoT network security through various IDS, this work uses XAI to leverage ML algorithms, also known as "black boxes," where the reasoning or logic behind the output predictions is not always clear. Because of their innate domain knowledge, human analysts continue to be essential for resource allocation and cybersecurity strategy development because this is the era of rising security issues concerning IoT network traffic. IDSs are needed to be modified and improved while understanding decisions taken by it.In this analysis, the UNSW-NB15 dataset is used as the basis for training a range of classifiers, including Random Forest, SVM and CatBoost. In addition, voting and stacking ensemble classifiers are employed for additional analysis. Subsequently, we employ XAI on those trained classifiers to explain the decisions displayed by those classifiers be viewed in a more comprehensible and detail-oriented manner, thereby facilitating human comprehension of the outcomes or decisions. XAI is employed and implemented using the python libraries ELI5 and LIME. This is how XAI was applied to the ML algorithms to assess its performance and this proceeds to demystify the IoT network intrusion detection. The proposed systems limitation may include the fact that the algorithms took a considerable amount of time to get trained and yield the results. Another limitation for this research work might be the models not keeping up with the ever-evolving dynamic nature of network intrusion attacks. The amount of new security breach attacks could overload the model, making it difficult for it to keep up with them. This paper's primary goal is to use XAI to assess the performance of ML algorithms applied. Therefore, it is limited that the emphasis is on XAI rather than the necessity to simplify the model's performance. Future research can incorporate other datasets apart from the UNSW-NB15 into the systems and use XAI to explain them to assess their performances.Enhancing the performance of the ML algorithms is another option. Moreover, alternative ML and DL algorithms as well as alternate XAI techniques like SHAP can be run on the balanced UNSW-NB15 dataset to achieve better interpretability and be considered as future scope.

References

[1] D. P. Möller and R. E. Haas, "Cybersecurity needs and benefits: The four rings model check for updates," in *Proceedings of International Conference on Information Technology and Applications: ICITA 2023*, vol. 839. Springer Nature, 2024, p. 461.

[2] R. G. Tiwari and M. Husain, "Explainable artificial intelligence (xai): Notions of explainability and its integration with blockchain," in *Convergence of Blockchain and Explainable Artificial Intelligence*. River Publishers, 2024, pp. 31–51.

[3] J. Sharma, M. Lal Mittal, G. Soni, and A. Keprate, "Explainable artificial intelligence (xai) approaches in predictive maintenance: A review," *Recent Patents on Engineering*, vol. 18, no. 5, pp. 18–26, 2024.

[4] M. Siganos, P. Radoglou-Grammatikis, I. Kotsiuba, E. Markakis, I. Moscholios, S. Goudos, and P. Sarigiannidis, "Explainable ai-based intrusion detection in the internet of things," in *Proceedings of the 18th International Conference on Availability, Reliability and Security*, 2023, pp. 1–10.

[5] S. B. Mallampati and S. Hari, "A review on recent approaches of machine learning, deep learning, and explainable artificial intelligence in intrusion detection systems," *Majlesi Journal of Electrical Engineering*, vol. 17, no. 1, pp. 29–54, 2023.

[6] H. Liu, C. Zhong, A. Alnusair, and S. R. Islam, "Faixid: A framework for enhancing ai explainability of intrusion detection results using data cleaning techniques," *Journal of network and systems management*, vol. 29, no. 4, p. 40, 2021.

[7] T. Dias, N. Oliveira, N. Sousa, I. Praça, and O. Sousa, "A hybrid approach for an interpretable and explainable intrusion detection system," in *International Conference on Intelligent Systems Design and Applications*. Springer, 2021, pp. 1035–1045.

[8] J. Ables, T. Kirby, W. Anderson, S. Mittal, S. Rahimi, I. Banicescu, and M. Seale, "Creating an explainable intrusion detection system using self organizing maps," in *2022 IEEE Symposium Series on Computational Intelligence (SSCI)*. IEEE, 2022, pp. 404–412.

[9] H. Lundberg, N. I. Mowla, S. F. Abedin, K. Thar, A. Mahmood, M. Gidlund, and S. Raza, "Experimental analysis of trustworthy in-vehicle intrusion detection system using explainable artificial intelligence (xai)," *IEEE Access*, vol. 10, pp. 102 831–102 841, 2022.

[10] S. K. Jagatheesaperumal, Q.-V. Pham, R. Ruby, Z. Yang, C. Xu, and Z. Zhang, "Explainable ai over the internet of things (iot): Overview, state-of-the-art and future directions," *IEEE Open Journal of the Communications Society*, vol. 3, pp. 2106–2136, 2022.

[11] A. Kumari and A. K. Mehta, "A hybrid intrusion detection system based on decision tree and support vector machine," in *2020 IEEE 5th International conference on computing communication and automation (ICCCA)*. IEEE, 2020, pp. 396–400.

[12] S. Neupane, J. Ables, W. Anderson, S. Mittal, S. Rahimi, I. Banicescu, and M. Seale, "Explainable intrusion detection systems (x-ids): a survey of current methods, challenges, and opportunities. arxiv," *Ithaca, NY*, 2022.

[13] O. Arreche, T. Guntur, and M. Abdallah, "Xai-ids: Toward proposing an explainable artificial intelligence framework for enhancing network intrusion detection systems," *Applied Sciences*, vol. 14, no. 10, p. 4170, 2024.

[14] A. Rehman, A. Farrakh, and S. Khan, "Explainable ai in intrusion detection systems: Enhancing transparency and interpretability,"

International Journal of Advanced Sciences and Computing, vol. 2, no. 1, pp. 7–20, 2023.

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

[16] S. More, M. Idrissi, H. Mahmoud, and A. T. Asyhari, "Enhanced intrusion detection systems performance with unsw-nb15 data analysis," *Algorithms*, vol. 17, no. 2, p. 64, 2024.

[17] A. Aleesa, M. Younis, A. A. Mohammed, and N. Sahar, "Deep-intrusion detection system with enhanced unsw-nb15 dataset based on deep learning techniques," *Journal of Engineering Science and Technology*, vol. 16, no. 1, pp. 711–727, 2021.

[18] M. Shantal, Z. Othman, and A. A. Bakar, "A novel approach for data feature weighting using correlation coefficients and min–max normalization," *Symmetry*, vol. 15, no. 12, p. 2185, 2023.

[19] C. Fan, M. Chen, X. Wang, J. Wang, and B. Huang, "A review on data preprocessing techniques toward efficient and reliable knowledge discovery from building operational data," *Frontiers in energy research*, vol. 9, p. 652801, 2021.

[20] S. Wali and I. Khan, "Explainable ai and random forest based reliable intrusion detection system," *Authorea Preprints*, 2023.

[21] A. Sadqui, M. Ertel, H. Sadiki, and S. Amali, "Evaluating machine learning models for predicting graduation timelines in moroccan universities," *International Journal of Advanced Computer Science and Applications*, vol. 14, no. 7, 2023.

[22] K. M. Ghori, R. A. Abbasi, M. Awais, M. Imran, A. Ullah, and L. Szathmary, "Performance analysis of different types of machine learning classifiers for non-technical loss detection," *IEEE Access*, vol. 8, pp. 16 033–16 048, 2019.

[23] M. Raihan-Al-Masud and H. A. Mustafa, "Network intrusion detection system using voting ensemble machine learning," in *2019 IEEE International Conference on Telecommunications and Photonics (ICTP)*. IEEE, 2019, pp. 1–4.

[24] S. Sivamohan and S. Sridhar, "An optimized model for network intrusion detection systems in industry 4.0 using xai based bi-lstm framework," *Neural Computing and Applications*, vol. 35, no. 15, pp. 11 459–11 475, 2023.