Agree to Disagree: Exploring Consensus of XAI Methods for ML-based NIDS

Katharina Dietz*, Mehrdad Hajizadeh†, Johannes Schleicher§, Nikolas Wehner*, Stefan Geißler*,
Pedro Casas‡, Michael Seufert§, Tobias Hoßfeld*
*University of Würzburg, Germany, †Technical University of Chemnitz, Germany
‡AIT Austrian Institute of Technology, Vienna, Austria, §University of Augsburg, Germany
*{katharina.dietz, nikolas.wehner, stefan.geissler, tobias.hossfeld}@uni-wuerzburg.de,
†mehrdad.hajizadeh@etit.tu-chemnitz.de, ‡pedro.casas@ait.ac.at, §{johannes.schleicher, michael.seufert}@uni-a.de

***Abstract*—The increasing complexity and frequency of cyber attacks require Network Intrusion Detection Systems (NIDS) that can adapt to evolving threats. Artificial intelligence (AI), particularly machine learning (ML), has gained increasing popularity in detecting sophisticated attacks. However, their potential lack of interpretability remains a significant barrier to their widespread adoption in practice, especially in security-sensitive areas. In response, various explainable AI (XAI) methods have been proposed to provide insights into the decision-making process. This paper investigates whether these XAI methods, including SHAP, LIME, Tree Interpreter, Saliency, Integrated Gradients, and DeepLIFT, produce similar explanations when applied to ML-NIDS. By analyzing consensus among these methods across different datasets and ML models, we explore whether an agreement exists that could simplify the practical adoption of XAI in cybersecurity, as similar explanations would eliminate the need for rigorous selection processes. Our findings reveal varying degrees of consensus among the methods, suggesting that while some align closely, others diverge significantly, highlighting the need for careful selection and combination of XAI tools to enhance trustworthiness in real-world applications.**

***Index Terms*—Machine Learning, Intrusion Detection, Explainable AI**

I. Introduction

The evolving data networks have transformed industries and everyday life, enabling massive communication, automation, and data exchange. This connectivity has increased the attack surfaces and posed more opportunities for attackers to infiltrate systems, leading to potential security breaches and financial losses. In a recent study, the European Union Agency for Cybersecurity (ENISA) reported substantial growth in cyber attacks with respect to their variety, number of incidents, and negative impacts [1]. Attackers increasingly leverage automation and artificial intelligence (AI) offensively [2] to enhance and sustain their malicious operations, continually adapting their malware to evade defense systems. Meanwhile, as adversaries improve their tactics, legacy security mechanisms alone, such as the signature-based methods (relying on known pre-defined attack signatures), cannot resist these sophisticated evolving threats, leaving systems vulnerable [3].

The advancement of AI and machine learning (ML) in various security domains, such as malware detection [4], advanced persistent threat (APT) detection [5], and network intrusion detection systems (NIDS) [6], has demonstrated the effectiveness of ML-based defenses in tackling evolving attack variants. NIDS, particularly those integrated with ML, play a crucial role in detection by learning underlying network traffic characteristics and continuously monitoring it to identify abnormal behavior. A large body of research is dedicated to build effective ML-NIDS, addressing hardware embedding [7], [8], scalability [9], hybrid approaches with signature-based methods [10], and improving ML model generalization [11].

Despite the numerous advantages that ML-NIDS offer, the explainability of AI models (XAI) remains challenging, causing security teams to be skeptical about adopting ML-NIDS in operational environments [6], [12]. The black-box nature of AI, combined with the lack of rational decision-making transparency, not only makes it challenging for security teams to understand detected suspicious events but also leaves AI-powered defense mechanisms vulnerable to adversarial attacks and information breaches [13], [14]. It is especially concerning in cybersecurity, where the implications extend beyond a cost-benefit analysis and could be expanded to serious issues, in some cases, even human lives [13]. Therefore, understanding how these ML algorithms make decisions is crucial for building trust, a principle that is highlighted in the European Union's General Data Protection Regulation (GDPR), known as the "right to explanation" for algorithmic decisions [15].

Thus, this paper is motivated by the observation that many studies have focused on achieving superior detection performance, but the explainability of these models is often falling short. In this study, we explore various state-of-the-art approaches to enhance the explainability of ML-NIDS outputs under different scenarios. As stated by Warnecke et al. [16], if all of these approaches generate similar explanations, any suitable method may be selected for practical adoption, which would eliminate the need for further selection criteria. Thus, we specifically zero in on the consensus among these methods, as this is a gap in current XAI research according to Krishna et al. [17]. Our main contributions are:

- A comparison of six XAI methods based on varying underlying AI/ML models, including traditional tree-based models, as well as (deep) neural networks.
- A consensus analysis among all valid combinations of explainers and models, in a ranked and unranked manner.
- Comprehensive qualitative and quantitative analyses, both

978-3-903176-66-9 ©2024 IFIP

local and global, were conducted to highlight the results across three distinct datasets.

This paper is structured as follows: Section III provides background information on intrusion detection and XAI, while Section III describes related works. Section IV outlines the proposed methodology for the consensus analyses and Section V presents the obtained results. Finally, Section VI summarizes the key findings and contributions of this study.

II. Background

Intrusion Detection Systems. In the context of cybersecurity, an intrusion involves any action that compromises the integrity, confidentiality, or availability of information or systems [18]. To detect and prevent such intrusions, Intrusion Detection Systems (IDS) and Intrusion Prevention Systems (IPS) have been developed. They can be deployed either directly on the host (HIDS), where they monitor local data, or in the network (NIDS), where they capture and analyze network traffic. For the latter, network traffic can be recorded on varying granularities, i. e., on a per-packet or per-flow basis (e. g., NetFlow/IPFIX). Due to the ever-evolving threat landscape [19], more and more attention has been paid to AI/ML-based NIDS in the past years [6]. While these models often achieve higher accuracies, their decision-making becomes less interpretable. So, in security-sensitive areas like NIDS, such black-box models may not actually be adopted in practice. Consequently, trust-enhancing techniques need to be adopted, especially regarding the European AI Act [20].

Explainable AI. Model explainability offers insights into the decision-making process and thus can enhance needed trust. This has led to the development of explainable AI (XAI), which consists of directly interpretable in-hoc/white box models like clustering, as well as post-hoc explainers. The latter analyze pre-trained models and are used either for local explanations of why the model makes a particular decision for a particular input, or for global explanations that return the overall important features [21]. Post-hoc explainers can be further classified into model-specific and -agnostic methods, as well as gradient- and perturbation-based approaches [22].

Perturbation-based explanation approaches are generally model-agnostic and create variations of the input data to see how the model's output changes in order to identify important features. Local Interpretable Model-agnostic Explanations (LIME) [23] focuses on sparse linear models as an explanatory model by learning the output of a model around a given input by generating various perturbations. The resulting model coefficients are then used as feature relevance values. Shapley Additive exPlanations (SHAP) [24] are based on the concept of Shapley values [25] from game theory, which was originally a concept to distribute a payout to players. Here, this concept attributes each feature a contribution to the model's prediction based on all possible subsets/combinations of features.

Gradient-based XAI approaches are explanatory methods that use the gradients of the model's output with respect to the input features to understand which features are influencing the prediction. Saliency maps [26] are generated by back-propagating the gradient of the output class with respect to the input through a trained Neural Network (NN). While originally developed for image classification tasks and Convolutional NNs (CNNs), Saliency maps can also be adapted for tabular data and other NN architectures. Integrated Gradients (IG) [27] considers the straightline path from an uninformative baseline (e. g., a black image or zero vector) to the input and computes the gradients for all points along the path. The single gradients are then integrated to determine the contribution of each input feature to the prediction. DeepLIFT [28] assigns feature importance by comparing activations from the actual input to a baseline and decomposes the output difference into contributions from each feature. This process is performed for each layer, with scores propagated forward through the network to determine the final feature contributions.

In addition to the above approaches, the Tree Interpreter (TI) [29] is a local model-specific explanation method for Decision Trees (DTs) or Random Forests (RFs) that breaks down predictions by calculating the impact of each split from root to leaf. The sum of these contributions, along with the average prediction, equals the final prediction.

III. Related Work

Research on NIDS has already explored XAI approaches, e. g., gradient-based methods were used to analyze important features in order to generate adversarial examples that can evade detection of current NIDS [30], whereas perturbation-based methods were extensively used to determine the most important features [21]. In this work, our goal is not to evaluate the different XAI methods in such an isolated manner, but rather to explore the consensus among them.

Warnecke et al. [16] evaluate six explanation methods for four malware datasets (i. e., malicious PDFs, apps, and code) using four Deep Learning (DL) models. Each DL model is dataset-specific. The results show that explainers that have insight on the model parameters like IG perform best. When such explainers are not feasible, LIME is recommended. In addition to other criteria, they also evaluate the consensus among those models by analyzing the intersection of top features and found that explanations differ and are therefore not interchangeable. Bhusal et al. [31] provide similar insights. In this work, we follow a similar approach by expanding this methodology, as well as applying it in the context of flow-based network traffic data. We explore both traditional ML and DL models in our analyses in a dataset-agnostic way.

Arreche et al. [32] already apply the methodology of Warnecke et al. on three NIDS datasets, but only evaluate the explanations of SHAP and LIME for seven ML models. Their results indicate that SHAP generally outperforms LIME which contradicts the previous results. While their work already includes some brief consensus evaluation on NIDS data among other criteria, it is only focused on the two perturbation-based approaches SHAP and LIME. In our work, we want to examine the agreement of more, and potentially more advanced, XAI approaches in more detail.

TABLE I: Overview of utilized datasets.

Name	Abbr.	Granularity	Size	Feats.[a]	#Attacks	%Anomal.
CIDDS-01	CIDDS	NetFlow	8 451 520	14	4	17.0%
CICIDS2017	CICIDS	Flow	692 703	78	5	36.5%
Edge-IIoTset	IIOT	Alerts+Logs	2 219 201	95	14	27.2%

[a] after encoding non-numerical features etc.

Tritscher et al. [22] examine that perturbation-based methods outperform gradient-based ones in explaining neural network behavior on tabular synthetic and real-world NIDS data. However, they also noted that the investigated explanatory models struggle with accurately explaining complex, non-linear models which indicates that the model choice depends on the dataset. The authors analyzed quantitatively and qualitatively in a subsequent work [33], how well SHAP works for three better functioning NIDS models. Hereby, they figured out that SHAP can provide high-quality explanations, while the choice of baseline has a high impact on the quality of the selected features. However, the choice of baseline is dependent on the model such that no overall preference could be provided. For our work, we choose a similar selection of XAI methods inspired by these two works. While the second work includes a consensus analysis regarding SHAP and three models, our work scales this up to a much broader range.

Hariharan et al. [34] compare global and local XAI techniques for IDS. They suggest that combining both approaches can balance detail with comprehensive understanding. They include a brief consensus analysis by investigating only the signs of the values for a single prediction (i. e., if a feature has a positive or negative impact on the explanation). In our work, we focus on the analyses in a quantitative manner and examine the actual rankings of features, not only their sign.

While there exist some works that analyze the consensus of XAI methods, our work extends on these works and investigates this challenge in a more detailed and comprehensive way, thereby, ultimately tying all these works together. We analyze five different ML models, including both shallow and deep models, and six explainers, including model-specific and model-agnostic explainers and also both perturbation- and gradient-based methods, on three different NIDS datasets. Additionally, we specifically focus on the consensus analyses.

IV. Methodology

NIDS Datasets. Table I depicts a summary of the three NIDS datasets we utilize in this work. The first dataset is the CIDDS-01 dataset [35]. It contains NetFlow data, which is one of the most common formats in real-world traffic monitoring. Thus, analyses based on such data are particularly relevant for practical situations. The dataset consists of various subsets which comprise different weeks and were measured at different vantage points. Here, we utilize the first week of this dataset. Besides benign traffic, the datasets contain records of Ping, Portscan, Bruteforce, and Denial-of-Service (DoS) attacks. Features include the standard NetFlow information, such as number of packets and bytes, as well as TCP flags.

The second dataset is the CICIDS2017 dataset [36]. While this dataset is also on a flow basis, the contained information is much richer than the information contained in typical

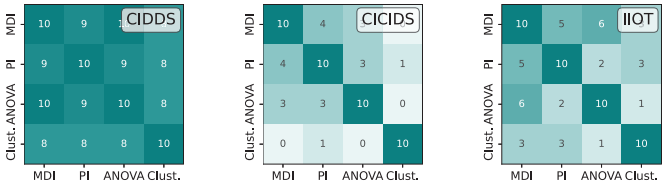


Fig. 1: Consensus heatmaps of different feature selection.

TABLE II: Compatible ML models and XAI methods.

	SHAP	LIME	TI	IG	DeepLIFT	Saliency
DT	✔	✔	✔	✘	✘	✘
RF	✔	✔	✔	✘	✘	✘
LGBM	✔	✔	✘	✘	✘	✘
SLP	✔	✔	✘	✔	✔	✔
MLP	✔	✔	✘	✔	✔	✔

NetFlow records. Additional features include, for example, various statistical moments about packet sizes and IATs. The dataset comprises five days, which all represent different attack scenarios. We utilize the Wednesday subset, which contains five types of DoS and Distributed DoS (DDoS) attacks. We choose this dataset, since as of today, it is one of the most utilized datasets in current NIDS research [6].

The third dataset is the Edge-IIoTset [37]. We chose this dataset since it is the most recent dataset and, in contrast to the previous two datasets, it also contains extra features extracted from alerts and log data. Features are derived from various IoT/Industrial IoT (IIoT) protocols, such as TCP, UDP, MQTT, MODBUS, and more. It contains 14 different attacks, including various types of DDoS attacks, Portscan, and more.

XAI Workflow. Firstly, the datasets are each split into 70% training and 30% test data and labels are binarized. Across all three datasets, we exclude features like IP addresses and ports to prevent overfitting on artifacts that attackers could spoof, ensuring the model reflects real-world scenarios. Zero-variance features are then filtered out, while the remaining features are min-max scaled. Lastly, we select the top ten features for each dataset, which directly impact the explanations. In the end, explanations can only be built on top of features that have been selected. The list of selected features and the full code for training and explaining is available online[1], which mainly makes use of scikit-learn [38], Captum [39], and PyTorch [40].

Figure 1 depicts the intersection of the top ten chosen features of four different selection methods, namely an impurity-based feature importance, which utilizes an RF to calculate the Mean Decrease in Impurity (MDI), Permutation Importance (PI) that randomly shuffles a feature's value and observes the impact on the model performance (here: also an RF), SelectKBest which utilizes an ANOVA-test, and lastly, a clustering-based approach that groups correlated features and chooses one representative out of each cluster. Each tile of the heatmaps depicts to which degree the four methods choose the same top ten features. For CIDDS, the shared features are relatively high, since we choose ten features out of 14. For the other two datasets, the differences are greater. In this work, we will utilize the traditional, impurity-based feature importance, since it also has the biggest overlap with the other methods.

[1] https://github.com/lsinfo3/cnsm2024-xai-nids-comparison

TABLE III: Model performance for different ML models across the three datasets.

	DT		RF		LGBM		SLP		MLP	
Dataset	$F1_{macro}$	$F1_{micro}$	$F1_{macro}$	$F1_{micro}$	$F1_{macro}$	$F1_{micro}$	$F1_{macro}$	$F1_{micro}$	$F1_{macro}$	$F1_{micro}$
CIDDS	0.9939	0.9965	0.9939	0.9966	0.9939	0.9965	0.9932	0.9962	0.9933	0.9962
CICIDS	0.9863	0.9872	0.9864	0.9873	0.9876	0.9885	0.9621	0.9648	0.9750	0.9766
EdgeIIoT	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000

TABLE IV: Most influential features[a] of explanations for all XAI methods for one random example of CIDDS.

DT			RF			LGBM		SLP					MLP				
SHAP	LIME	TI	SHAP	LIME	TI	SHAP	LIME	SHAP	LIME	Sal.	IG	DLIFT	SHAP	LIME	Sal.	IG	DLIFT
S	S	Dur.	S	S	S	S	S	S	S	Dur.	S	S	F	Tos	Dur.	F	F
Dur.	Tos	S	F	Tos	F	Dur.	F	F	Tos	Tos	F	F	S	UDP	Tos	S	S
P	F	Tos	P	Dur.	Pkts	F	Tos	Tos	F	Pkts	Pkts	Pkts	Tos	R	Pkts	Pkts	Pkts
UDP	P	F	Dur.	P	Dur.	UDP	Dur.	Dur.	UDP	S	A	A	P	F	R	TCP	A
Tos	Dur.	P	Pkts	UDP	P	Tos	R	P	Dur.	R	TCP	TCP	UDP	P	UDP	A	TCP

[a] The features A, S, F, P, R depict the TCP Flags ACK, SYN, FIN, PSH, RST, respectively; Tos = Type of service in the IP header, defines packet priority.

Since our goal is to not only compare the explanations of different XAI methods but also the impact of varying underlying models, we select a variety of ML models. Tree-based models and NNs constitute the most popular ML models in recent IDS literature [6]. Overall, we select five models; three tree-based ones (RF, DT, Light Gradient-Boosting Machine (LGBM)), and two NNs, namely a simple Single-Layer-Perceptron (SLP) and a Multi-Layer-Perceptron (MLP). Table II illustrates the compatibility of these models with the chosen explainers. In total, we select six explainers, namely SHAP, LIME, Tree Interpreter (TI), Saliency, Integrated Gradients (IG), and DeepLIFT, giving us a mixed selection of gradient- and perturbation-based XAI methods. The table shows that all five models are compatible with SHAP[2] and LIME. TI is only implemented for two of our models (RF, DT). Lastly, IG, DeepLIFT, and Saliency only work for NNs.

Consensus Quantification. After selecting our ML models and their corresponding explainers, the next step is to determine how to calculate the consensus across the different model+explainer combinations to measure their agreement or disagreement. For this, we employ a sign, an unranked, and a ranked quantification, as similar metrics have been found useful by Krishna et al. [17] in other domains, such as finances, images, and texts based on interviews with data scientists.

For the *sign quantification*, we simply look at the signs (positive, negative, or zero) of assigned impact values for *all* features, i. e., if the feature has a positive or negative impact on the decision w.r.t. to the explanation for the targeted class. Note that a negative impact does not mean that a feature worsens the model's performance or that this feature is unimportant. This consensus should ideally be better than a random assignment. Roughly 33% of all signs would match when randomly guessing, since we also account for zeros.

For the *unranked quantification*, we look at the five most influential features (out of the ten chosen features) for each of the resulting explanations and calculate the intersection of them without respect to the ranking of these top five features. Ideally, this should be better than a random explanation, i. e., a random selection of features. In other words, assume that we have a given explanation by one of the explainers. If we now draw at random five features from the total number of features, the expected random unordered consensus follows a hypergeometric distribution. This distribution has an expected value of $n\frac{M}{N}$, where n is the number of draws (five here), M is the number of elements with the "correct" characteristic (in our case also five, since we want to match the five features from the given explanation), and N is the number of total elements (number of total features for us). For example, analyzing the consensus of the top five features out of ten total features with a random feature selection has an expected consensus of 2.5 features. This serves as a baseline comparison, to quantify if the degree of agreement of the different explanations is simply due to random matching.

In the *ranked quantification* we now take the actual order of the features in the explanations into account. Specifically, we are interested in how many of the top features match in order, e. g., only the first feature matches, the first two, and so on. This also means that we are not interested if, e. g., only the fourth feature matches if the ones before do not. Again, we want to have a better consensus than a random ordering of the features. Sticking with the example for ten total features, the probability for zero matches is $\frac{9}{10}$, since must not draw the top feature at random from the ten. The probability for exactly one match is $\frac{1}{10} \cdot \frac{8}{9}$, since we have a 1 in 10 chance to randomly draw the top feature. Then we have to avoid drawing the second most important feature to have exactly one match. In general, the probabilities for having exactly k matches when randomly drawing M features are defined as follows:

$$P(k) = \begin{cases} \frac{(N-k)!}{N!} \cdot \frac{N-k-1}{N-k} & (\text{if } k < M) \\ \frac{(N-k)!}{N!} & (\text{if } k = M) \end{cases}$$

Note that the probability for M matches is missing the last factor, since we only look at the top M features, and any match that would happen afterward is irrelevant. Thus, the expected random ranked consensus, i. e., the expected value of the number of matches when M features are randomly drawn is $E[X] = \sum_{k=0}^{M} k \cdot P(k)$. In our case of comparing the consensus of the five top features out of ten total features, the expected random consensus is around 0.113 features, which will serve as the baseline for the analyses below.

V. Evaluation

Preliminary Performance Analysis. Before diving into the XAI-related analyses, we first want to briefly examine the actual model performance. This ensures that the (dis)agreement of the various combinations of models and explainers is not

[2] Note that we use the *Tree-* and *DeepExplainer* here, which are actually tailored to the specific models to efficiently **approximate** the SHAP values.

a byproduct of underperforming classifiers. For this, Table III depicts the performance of all five chosen ML models on all three datasets via the macro and micro F1-scores. All models achieve high scores, most of them exceeding 98%, with the exception of the NN-based models for the CICIDS dataset, where the performance slightly drops, but still remains high, nonetheless. Since we look at the macro F1, we also account for the imbalances in the datasets.

Exemplary Local Qualitative Analysis. For our actual XAI analysis, we follow a bottom-up approach, by first investigating a local explanation more qualitatively, before looking into local and global explanations in a more quantitative manner. To explain how the output of the different explainers looks like (i. e., the most influential features), Table IV depicts the explanations of why the models chose the predicted class for all valid combinations of model+explainer and for a random example from the test set of the CIDDS dataset. Cyan-colored cells illustrate features that positively impact the decision, while red-colored cells depict features that negatively impact the decision. The vibrancy of the colors depicts the respective feature's overall influence w.r.t. to the most important feature, i. e., darkred and darkcyan both have a decisive impact, while lighter colors make less of a difference.

Overall, the table illustrates that – for this specific sample – 11 out of 18 approaches agree at least that the SYN flag is the topmost positively impactful feature. Starting with the second most important feature, the opinions already diverge w.r.t. to the relative strength of the impact, as well as the overall ranking of features. This happens within the same XAI method, as well as for the same model, especially for the SLP and MLP. There are also some contradictions in the feature's signs, e. g., while LIME assigns the RST flag a positive influence for MLP, it assigns the topmost negative influence for LGBM. In summary, while there is a trend of agreement regarding the top features, there are already some notable disagreements visible, even within similar ML models.

Local Quantitative Analysis. Since we only investigated a single exemplary explanation regarding the overall consensus, we now shift our view to a more aggregated analysis. For this, Figure 2 depicts three types of heatmaps over all 18 possible combinations of model+explainer, illustrating the sign, unranked, and ranked consensus of for all three datasets for 1k randomly selected samples from the test data. The more vibrant the color, the higher the consensus.

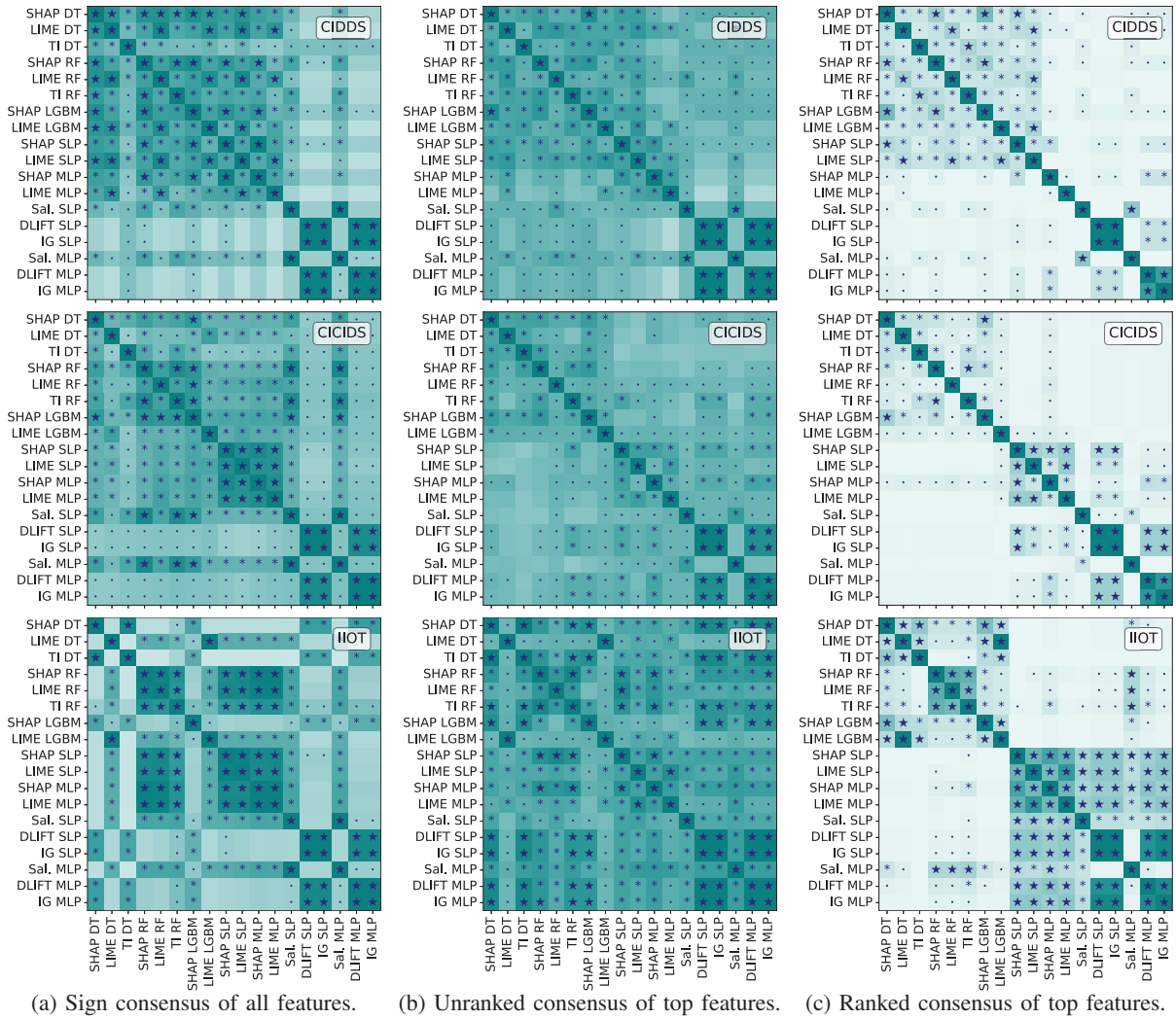
For the *sign consensus*, the small stars mark an average consensus of 50% to 75%, and the big stars mark an average consensus of greater than 75% of all ten chosen features. The baseline value (33%) is marked by small dots. For CIDDS, we see a clear distinction between the permutation-based methods and gradient-based methods. In these two groups, the agreement of methods is high as indicated by the stars. Between the groups, however, the consensus is often lower than the baseline, meaning that the methods do not even agree on a sign. For CICIDS, we see a similar trend. However, in the group of permutation-based methods, we now see a clearer trend of explainers with a similar model agreeing more compared to others, e. g., SLP and MLP, and for almost all cases the consensus is higher than the baseline. Lastly, for IIOT, the consensus here is more intermixed among the different methods. Looking at the actual explanations and their impact, on this dataset already a few features are sufficient for a good classification. This eases the overall agreement of diverging methods since the remaining features are not as important. Though, compared to the other datasets, there are more gaps, as these remaining features potentially just get assigned a value of '0' in terms of impact for some methods, while other methods at least assign a small contribution.

For the *unranked consensus*, the small stars mark an average consensus between three and four features, the big stars mark an average consensus of greater than four of the top five features (out of ten chosen features). The baseline value (2.5) is marked by small dots. For CIDDS, we see a similar trend to the sign consensus, where there is a distinction between the two general groups of explainers. For CICIDS, however, the trend diverges now. We can only see a faint trend that similar models (tree-based and NN-based) now agree more. For IIOT, the overall consensus is higher, since the distinction between normal and attack traffic is enabled by only a few features, similar to the sign analysis. For all datasets, the unranked consensus is mostly at least on par with random guessing.

For the *ranked consensus*, the small stars mark an average consensus between 0.5 and one feature(s), and the big stars mark an average consensus of greater than one of the top five features (out of ten chosen features). The baseline value (0.113) is marked by small dots. In general, the consensus is overall lower compared to the unranked analysis, which is expected, since this consensus is much harder to achieve. For many methods, their explanations do not even match on the most important feature, which is also showcased by the fact in many cases the consensus is even worse than a random guess.

In summary, all three consensus analyses showed, that while there are some trends towards agreement – especially between perturbation- vs. gradient-based methods and when the same underlying model is used – XAI approaches for NIDS are not generally interchangeable, and even similar model+explainer combinations struggle to consistently agree on the top one feature only. For example, for IIOT and CICIDS, the DT and RF barely even agree on the first feature. The trends where approaches agree or not are also inconsistent, i. e., the gaps in the consensus heatmaps are dependent on the dataset.

Global Quantitative Analysis. In addition to the local analyses, we can sum up the absolute values of each instance to get an overall score per method. While explainers are primarily designed for individual explanations, this aggregated analysis may offer insights on consensus at a global level (but may overlook local nuances). Table V illustrates this for CIDDS only for brevity's sake, where a darker color depicts a high impact feature (color gradient w.r.t. to top 1 feature). We see a correlation between the amount of consensus and type of explainer, i. e., IG and DeepLIFT agree, SHAP and LIME agree (partially also with TI), and Saliency is a bit of an outlier. Saliency assigns the flow duration the highest impact by far,



(a) Sign consensus of all features. (b) Unranked consensus of top features. (c) Ranked consensus of top features.

Fig. 2: Comparison heatmaps of XAI consensus for the three different datasets.

TABLE V: Global analysis[a] for the CIDDS dataset.

SHAP					LIME					TI		Saliency		IG		DeepLIFT	
DT	RF	LGBM	SLP	MLP	DT	RF	LGBM	SLP	MLP	DT	RF	SLP	MLP	SLP	MLP	SLP	MLP
S	S	S	S	UDP	S	S	S	S	Tos	S	S	Dur.	Dur.	TCP	A	TCP	TCP
Dur.	Dur.	Dur.	Tos	Tos	Tos	Tos	F	F	UDP	Dur.	Dur.	Pkts	Pkts	A	TCP	A	A
A	F	F	UDP	F	F	UDP	R	Tos	R	Tos	Pkts	Tos	Tos	S	F	S	Tos
UDP	Pkts	A	F	P	UDP	Dur.	Tos	UDP	F	R	F	R	UDP	F	Tos	F	F
Tos	Tos	Tos	A	A	Pkts	Pkts	Dur.	Dur.	P	Pkts	Tos	UDP	R	UDP	S	UDP	S

[a] The features A, S, F, P, R depict the TCP Flags ACK, SYN, FIN, PSH, RST, respectively; Tos = Type of service in the IP header, defines packet priority.

while other methods assign it a lower impact or do not consider this for their top five at all. Interestingly, Saliency also does not consider the rest of the features as important, which is another reason for the disagreement between Saliency and the other methods, since these features may potentially be more random when looking at the top five. This global view directly explains the heatmap trends on the basis of the actual features.

For CICIDS, the global insights shed some light onto why the trends for the unranked analysis might be more faint, but also more intermixed w.r.t. to perturbation- vs. gradient-based methods, compared to CIDDS. The top features of all methods are often related to the packet sizes. So, the explanations in itself are actually quite similar among the methods. Some methods choose the mean packet size, some the maximum, and some only focus on one flow direction for their top 1.

Lastly, for IIOT, the global analysis reveals that many top features are related to the MQTT protocol (name, flags, or topic). Many methods even consider these features by far the most important. This explains why the consensus is generally higher here, as already briefly mentioned for the sign consensus. Moreover, this also explains why the consensus is divided into smaller groups, since it depends which one of the features they rank as the top feature (and in some cases assign the rest no/less importance), even though they are all suitable.

In summary, the global analysis reveals the reasoning behind the trends of the heatmaps. It also shows that one potential reason behind a lower consensus could be due to correlations between features, making multiple explanations valid. Ultimately, this highlights the need to not only carefully select different XAI methods but also to keep the their input concise.

VI. Conclusion

In this work, we explored the consensus among various explainable AI (XAI) methods for Machine Learning (ML)-based Network Intrusion Detection Systems (NIDS). Given the variety of XAI techniques, such as SHAP, LIME, Tree Interpreter (TI), Saliency, Integrated Gradients (IG), and DeepLIFT, an important question is whether these methods provide consistent explanations for NIDS. If a strong consensus exists, it could reduce the need for rigorous individual evaluations and ease the practical adoption of ML-based NIDS.

Our findings indicate that not all XAI methods are interchangeable for ML-NIDS. Depending on the dataset, perturbation- and gradient-based methods diverged, while in other cases the underlying model was more important. The gaps in consensus are overall inconsistent and dataset-dependent. Even similar models match in some cases not better than random guesses, and many approaches do not even agree on the top 1 feature consistently. Our analyses also revealed that while explanations may diverge in terms of exact features, they might choose related or correlated features instead, making multiple explanations valid. For network monitoring and NIDS specifically, this highlights the importance of not overloading models with unnecessary information about the network traffic, since the explainers can only work with what they are given. Due to the found gaps in consensus, it seems reasonable to not rely solely on a single explanation, e. g., by utilizing multiple XAI approaches [41]. In this case, removing correlations between features seems necessary to avoid having different explainers present alternative but equally valid explanations, which further reduces consensus. Alternatively, we can soften the metrics by taking the weights of the features into account, instead of just ordinal rankings [17].

In future work, we aim to explore multi-label classification, as well as evaluating class-specific explanations to better understand how attacks vs. normal traffic are interpreted by XAI methods, since in this work we looked at randomly selected samples without respect to the classes. Instead of utilizing the already preprocessed datsets, we also want to unify feature sets among different datasets by extracting own custom features, preferably on real-world/more realistic traffic. We also plan to extend our work with a parameter study on alternative feature selection methods and varying number of features, to quantify how we can affect the consensus positively.

Acknowledgment

This work has been funded by the Bavarian Ministry of Economics, Regional Development and Energy (StMWI) as part of the project VIPNANO (DIK-2307-0006) and by Deutsche Forschungsgemeinschaft (DFG) under grant SE 3163/3-1, project nr.: 500105691 (UserNet). The authors alone are responsible for the content.

References

[1] C. Ardagna, S. Corbiaux, K. Van Impe, and R. Ostadal, "Enisa threat landscape 2023," *European Union Agency for Cybersecurity (ENISA)*, 2023.

[2] F. N. Motlagh, M. Hajizadeh, M. Majd, P. Najafi, F. Cheng, and C. Meinel, "Large language models in cybersecurity: State-of-the-art," *arXiv preprint arXiv:2402.00891*, 2024.

[3] L. Caviglione, M. Choraś, I. Corona, A. Janicki, W. Mazurczyk, M. Pawlicki, and K. Wasielewska, "Tight arms race: Overview of current malware threats and trends in their detection," *IEEE Access*, 2020.

[4] D. Ucci, L. Aniello, and R. Baldoni, "Survey of machine learning techniques for malware analysis," *Computers & Security*, 2019.

[5] M. A. Talib, Q. Nasir, A. B. Nassif, T. Mokhamed, N. Ahmed, and B. Mahfood, "APT beaconing detection: A systematic review," *Computers & Security*, 2022.

[6] K. Dietz, M. Mühlhauser, J. Kögel, S. Schwinger, M. Sichermann, M. Seufert, D. Hermann, and T. Hoßfeld, "The missing link in network intrusion detection: Taking AI/ML research efforts to users," *IEEE Access*, 2024.

[7] B. Brandino, E. Grampin, K. Dietz, N. Wehner, M. Seufert, T. Hoßfeld, and P. Casas, "HALIDS: A hardware-assisted machine learning IDS for in-network monitoring," in *TMA*, 2024.

[8] P. Golchin, C. Zhou, P. Agnihotri, M. Hajizadeh, R. Kundel, and R. Steinmetz, "CML-IDS: Enhancing intrusion detection in SDN through collaborative machine learning," in *IEEE CNSM*, 2023.

[9] M. Seufert, K. Dietz, N. Wehner, S. Geißler, J. Schüler, M. Wolz, A. Hotho, P. Casas, T. Hoßfeld, and A. Feldmann, "Marina: Realizing ML-driven real-time network traffic monitoring at terabit scale," *IEEE TNSM*, 2024.

[10] Z. Chiba, N. Abghour, K. Moussaid, A. E. Omri, and M. Rida, "Newest collaborative and hybrid network intrusion detection framework based on Suricata and isolation forest algorithm," in *SCA*, 2019.

[11] P. Golchin, N. Rafiee, M. Hajizadeh, A. Khalil, R. Kundel, and R. Steinmetz, "SSCL-IDS: Enhancing generalization of intrusion detection with self-supervised contrastive learning," in *IFIP/IEEE Networking*, 2024.

[12] G. Apruzzese, P. Laskov, and J. Schneider, "SoK: Pragmatic assessment of machine learning for network intrusion detection," in *IEEE EuroS&P*, 2023.

[13] Z. Zhang, H. Al Hamadi, E. Damiani, C. Y. Yeun, and F. Taher, "Explainable artificial intelligence applications in cyber security: State-of-the-art in research," *IEEE Access*, 2022.

[14] G. Jaswal, V. Kanhangad, and R. Ramachandra, *AI and deep learning in biometric security: trends, potential, and challenges*. CRC Press, 2021.

[15] B. Goodman and S. Flaxman, "European union regulations on algorithmic decision-making and a "right to explanation"," *AI magazine*, 2017.

[16] A. Warnecke, D. Arp, C. Wressnegger, and K. Rieck, "Evaluating explanation methods for deep learning in security," in *IEEE EuroS&P*, 2020.

[17] S. Krishna, T. Han, A. Gu, J. Pombra, S. Jabbari, S. Wu, and H. Lakkaraju, "The disagreement problem in explainable machine learning: A practitioner's perspective," *arXiv preprint arXiv:2202.01602*, 2022.

[18] T. Grance, J. Hash, M. Stevens, K. O'Neal, and N. Bartol, *Guide to information technology security services*. National Institute of Standards and Technology, Technology Administration, US Department of Commerce, 2003.

[19] "Cisco cybersecurity readiness index," Tech. Rep., March 2024. [Online]. Available: https://newsroom.cisco.com/c/r/newsroom/en/us/a/y2024/m03/cybersecurity-readiness-index-2024.html

[20] https://eur-lex.europa.eu/eli/reg/2024/1689, accessed: 2024-08-30.

[21] M. Wang, K. Zheng, Y. Yang, and X. Wang, "An explainable machine learning framework for intrusion detection systems," *IEEE Access*, 2020.

[22] J. Tritscher, M. Ring, D. Schlör, L. Hettinger, and A. Hotho, "Evaluation of post-hoc XAI approaches through synthetic tabular data," in *ISMIS*, 2020.

[23] M. T. Ribeiro, S. Singh, and C. Guestrin, ""Why should I trust you?": Explaining the predictions of any classifier," 2016. [Online]. Available: https://arxiv.org/abs/1602.04938

[24] S. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," 2017. [Online]. Available: https://arxiv.org/abs/1705.07874

[25] L. S. Shapley, "A value for n-person games," *Contributions to the Theory of Games*, 1953.

[26] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep inside convolutional networks: Visualising image classification models and saliency maps," *CoRR*, vol. abs/1312.6034, 2013. [Online]. Available: https://api.semanticscholar.org/CorpusID:1450294

[27] M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic attribution for deep networks," *CoRR*, vol. abs/1703.01365, 2017. [Online]. Available: http://arxiv.org/abs/1703.01365

[28] A. Shrikumar, P. Greenside, and A. Kundaje, "Learning important features through propagating activation differences," in *ICML*, 2017.

[29] A. Saabas, "Interpreting random forests," 2018, accessed: 2024-08-14. [Online]. Available: http://blog.datadive.net/interpreting-random-forests/

[30] H. Qiu, T. Dong, T. Zhang, J. Lu, G. Memmi, and M. Qiu, "Adversarial attacks against network intrusion detection in iot systems," *IEEE IoT-J*, 2021.

[31] D. Bhusal, R. Shin, A. A. Shewale, M. K. M. Veerabhadran, M. Clifford, S. Rampazzi, and N. Rastogi, "SoK: Modeling explainability in security analytics for interpretability, trustworthiness, and usability," in *ACM ARES*, 2023.

[32] O. Arreche, T. R. Guntur, J. W. Roberts, and M. Abdallah, "E-XAI: Evaluating black-box explainable AI frameworks for network intrusion detection," *IEEE Access*, 2024.

[33] J. Tritscher, M. Wolf, A. Hotho, and D. Schlör, "Evaluating feature relevance XAI in network intrusion detection," in *xAI*, 2023.

[34] S. Hariharan, R. Rejimol Robinson, R. R. Prasad, C. Thomas, and N. Balakrishnan, "XAI for intrusion detection system: comparing explanations based on global and local scope," *Journal of Computer Virology and Hacking Techniques*, 2023.

[35] M. Ring, S. Wunderlich, D. Grüdl, D. Landes, and A. Hotho, "Flow-based benchmark data sets for intrusion detection," in *ECCWS*, 2017.

[36] I. Sharafaldin, A. H. Lashkari, A. A. Ghorbani *et al.*, "Toward generating a new intrusion detection dataset and intrusion traffic characterization." *ICISSp*, vol. 1, pp. 108–116, 2018.

[37] M. A. Ferrag, O. Friha, D. Hamouda, L. Maglaras, and H. Janicke, "Edge-IIoTset: A new comprehensive realistic cyber security dataset of IoT and IIoT applications for centralized and federated learning," *IEEE Access*, 2022.

[38] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg *et al.*, "Scikit-learn: Machine learning in Python," *JMLR*, 2011.

[39] N. Kokhlikyan, V. Miglani, M. Martin, E. Wang, B. Alsallakh, J. Reynolds, A. Melnikov, N. Kliushkina, C. Araya, S. Yan *et al.*, "Captum: A unified and generic model interpretability library for PyTorch," *arXiv preprint arXiv:2009.07896*, 2020.

[40] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "PyTorch: An imperative style, high-performance deep learning library," *NeurIPS*, 2019.

[41] C. Pirie, N. Wiratunga, A. Wijekoon, and C. F. Moreno-Garcia, "AGREE: A feature attribution aggregation framework to address explainer disagreements with alignment metrics." CEUR-WS, 2023.