Bridging the Explanation Gap in AI Security: A Task-Driven Approach to XAI Methods Evaluation

Ondrej Lukas[a] and Sebastian Garcia[b]
Department of Computer Science, Faculty of Electrical Engineering, Czech Technical University in Prague, Czech Republic

Keywords: Explainable AI, Functional Metrics, Explanation Evaluation, Network Security.

Abstract: Deciding which XAI technique is best depends not only on the domain, but also on the given task, the dataset used, the model being explained, and the target goal of that model. We argue that the evaluation of XAI methods has not been thoroughly analyzed in the network security domain, which presents a unique type of challenge. While there are XAI methods applied in network security there is still a large gap between the needs of security stakeholders and the selection of the optimal method. We propose to approach the problem by first defining the stack-holders in security and their prototypical tasks. Each task defines inputs and specific needs for explanations. Based on these explanation needs (e.g. understanding the performance, or stealing a model), we created five XAI evaluation techniques that are used to compare and select which XAI method is best for each task (dataset, model, and goal). Our proposed approach was evaluated by running experiments for different security stakeholders, machine learning models, and XAI methods. Results were compared with the AutoXAI technique and random selection. Results show that our proposal to evaluate and select XAI methods for network security is well-grounded and that it can help AI security practitioners find better explanations for their given tasks.

1 INTRODUCTION

Explaining the different parts of the machine learning pipeline (XAI) is critical for many tasks, such as if a model is trusted, or knowing which parts of a model to modify to improve it (Arrieta et al., 2019). The explanation methods should also be evaluated to understand their properties.

The evaluation methods for XAI (e.g. stability and robustness) help understand if the XAI is aligned with the decision task. While some work is done for such evaluation in other domains, they seem insufficient in the network security domain.

The network security domain presents unique characteristics that make adapting current XAI evaluation methods problematic (Jacobs et al., 2022). In particular, this domain suffers from three problems: (i) lack of good datasets, (ii) imbalance (Sacher, 2020), and (iii) high costs of errors (Amit et al., 2019). With some datasets having almost a 1:100000000 ratio of attack-to-benign packets, even the better false positive rates need to be in the order of 0.001% to be usable. The majority of available datasets are synthetic or represent too small networks. Additionally, privacy concerns and legal issues forbid many datasets from being available or verified.

Evaluating XAI methods, in general, is difficult because there is no unique correct result (van der Waa et al., 2021; Sokol and Flach, 2020). The majority of existing work therefore focuses on the evaluation of a single property of the explanation. OpenXAI (Agarwal et al., 2022) proposes several metrics for XAI method comparison while AutoXAI (Cugny et al., 2022) aims to automate both the method selection process and hyperparameter tuning. However, the user input is required in the form of desired properties of the explanation. Neither of the methods includes any security-related dataset in their comparisons.

In the cybersecurity domain, XAI depends also on the security tasks (Warnecke et al., 2020); which are representative of typical cybersecurity roles. When XAI is applied to these security tasks, it is seen that the traditional evaluation methods of XAI may fall short of the desired cybersecurity needs.

Given the impact of machine learning (ML) models in network security (Aledhari et al., 2021), there is an increasing need for verified XAI explanations to

[a] https://orcid.org/0000-0002-7922-8301
[b] https://orcid.org/0000-0001-6238-9910

Lukas, O. and Garcia, S.
Bridging the Explanation Gap in AI Security: A Task-Driven Approach to XAI Methods Evaluation.
DOI: 10.5220/0012475200003636

In *Proceedings of the 16th International Conference on Agents and Artificial Intelligence (ICAART 2024) - Volume 3*, pages 1370-1377
ISBN: 978-989-758-680-4; ISSN: 2184-433X

enhance transparency, comprehension, trust, ethics, and cooperation between humans and AI systems in network security. Therefore, a correct understanding of the properties of network security, XAI, and their evaluations is beneficial for the community.

This paper proposes a method to select the best-suited XAI method for each specific network security task according to the conditions and restrictions of these tasks, the datasets, and the ML model used. Our proposal understands the most common tasks for the network security stakeholders, then identifies the exact evaluation properties they need in their explanations, and uses these evaluation techniques in state-of-the-art XAI methods. Finally, our proposed method chooses the XAI that best explains the selected security task to the stakeholder. A key part of our technique is the exploration of the dependence of the evaluation and XAI methods the dataset-model-goal triplet.

Our method first identifies four tasks that represent archetypal security stakeholders: model extraction, model improvement, model understanding, and single sample understanding. Each task has a specific security goal and restrictions. The tasks differ not only in the goal but also in the access to the model and both training and testing data. Lastly, we propose a scoring technique to evaluate and compare the performance of XAI algorithms in an absolute way to answer the question, 'How good was this XAI to explain this security task?'. Finally, our method chooses the best-suited XAI algorithm.

In our experiments, we use four ML models (Random Forest, Gradient Boosting Trees, Multi-layer Perceptron, and Support Vector Machines) and four XAI methods (SHAP, LIME, Anchors, and RISE) for each of the proposed tasks.

The performance of an ML model generally depends on the data it has been trained on and on the goal that has been optimized to perform (Sarker, 2021); therefore, the performance of an XAI algorithm at least also depends on the model, the data, and the optimized goal. Therefore, we include a random explanation baseline in our evaluation to understand if the differences in XAI methods are significant and to what extent.

Results show that selecting the best XAI method highly depends on the task, the model to explain, and the dataset chosen, confirming previous work and hypotheses. Given the variability of the models and data, we did not even find an XAI method that consistently outperforms others. We conclude that XAI techniques should always be evaluated concerning the triplet of task-model-data because their power of explanation should be aligned with the needs of the stakeholders of those tasks. Also, the specific evaluation technique plays a crucial role in selecting XAI methods, and stakeholders should consider their capabilities when choosing according to the domain.

The contributions of this paper are:

- An identification of four main tasks of network security stakeholders needs.
- A new evaluation technique of XAI methods for scoring and comparing XAI algorithms based on adversarial approaches.
- An evaluation on four ML algorithms in a cybersecurity task and four XAI algorithms.
- Use of a real-world labeled network security dataset for XAI evaluation.

2 RELATED WORK

The widely accepted approach to the evaluation of explanation originally proposed in (Doshi-Velez and Kim, 2017) splits the evaluation approaches into categories according to human involvement in the evaluation process.

The variety of needs for XAI recipients is examined from the point of goal (Arrieta et al., 2020), audience type (stakeholders) (Nadeem et al., 2022; Doshi-Velez and Kim, 2017) and specific needs, concerns, and goals of the recipients of the explanations (Jin et al., 2023) and their domain knowledge (Nguyen et al., 2020).

Among the most widely used functional metrics for assessing the quality of the explanations are **Faithfulness** (Alvarez-Melis and Jaakkola, 2018)(also referred to as fidelity(Yeh et al., 2019)) which is a measurement of how much the explanation approximates the prediction of the model that is being explained and **Stability** (Krishna et al., 2022) or Robustness (Alvarez-Melis and Jaakkola, 2018) are metrics that evaluate the simple assumption of explanations:*similar inputs should give rise to similar explanations*. Both metrics are computed by measuring how much the explanations change when there are small perturbations to the input, model representation, or output of the underlying predictor. **Compactness** of the explanation is another desirable property (Mothilal et al., 2021). In cases, such as rule or example sets, we measure the cardinality of those sets. Other approaches limit the size/length of the explanation and measure how well it covers the space of input samples (Schwalbe and Finzel, 2023).

Explainable Security (Vigano and Magazzeni, 2020; Parmar, 2021) establishes the field of explainable security, proposing stakeholder differentiation

and important questions for building and evaluating XAI for security.

Nadeem et al.(Nadeem et al., 2022) provide a summary of notable work in both XAI methods in security and their evaluation. It shows that only 1-in-5 applications of XAI in security were accompanied by any kind of evaluation of the explanations. A combination of general and security-related criteria is proposed by Warnecke et al. (Warnecke et al., 2020). Additionally, the authors show that XAI methods are unsuitable for general usage in security, most notably GuidedBackProp and GradCam (GradCAM++).

The landscape of the explainable methods is diverse and creating a meaningful taxonomy is not a simple task. We adopt the categorization first proposed by Molnar (Molnar, 2022) and extended by in (Schwalbe and Finzel, 2023) by introducing a comprehensive taxonomy of XAI based on several criteria such as *input*, *output*, *form of presentation* or *interactivity* of the methods.

Currently, the most adopted explanation methods are (i) interpretable surrogates (decision trees, linear models), (ii) rules, and (iii) model-agnostic methods. In the former category, methods focusing on the importance of input features such as SHAP (Lundberg and Lee, 2017), LIME (Ribeiro et al., 2016) or RISE (Petsiuk et al., 2018) are widely used. Another approach is explaining models by examples - prototypes (Ribeiro et al., 2018; Kim et al., 2016), or counterfactuals (Wachter et al., 2017).

Historically, the majority of the XAI methods were created in other domains (such as computer vision). While adapting such methods to network security is somewhat possible, shown in (Nadeem et al., 2022), the specifics of the target domain are not considered for most of the XAI methods.

3 METHODOLOGY

Our work aims to select the best XAI method to explain a model in a specific task based on a particular **goal** of the task, the **ML model** to be explained, and the **dataset**. In the context of this paper, the goal is represented by a task for which the XAI method can be used, such as *Model output understanding* or *Model Improvement*. The suitability of an XAI method for a given task and ML model is measured by the level of fulfillment of the task. The metrics differ in each task and are described in subsections 3.1.

Our methodology consists of (i) defining four security tasks and corresponding metrics for measuring the level of fulfillment of the test; (ii) training 4 ML methods on a dataset of a real network security problem (detection of attacks only looking at encrypted TLS traffic); Applying the selected XAI methods (SHAP, LIME, Anchor and RISE) to defined tasks and computing the metrics; (v) select the best XAI for the task-model-data triplet.

Unlike previously proposed methods, our evaluation does not require Ad Hoc selection of metrics and their importance for a given task which requires deep understanding on the side of the user. Functional metrics such as robustness or fidelity evaluate purely the relation between the data point and the explanation or model output and the evaluation, not the impact on the goal, being solved with the XAI method.

3.1 Network Security Tasks

We identified four common network security stakeholders: the final user, the model creator, the model evaluator, and the adversarial attacker. Each of these stakeholders has different needs, domain knowledge, and available assets. The end-user commonly does not have full access to the ML model and only has a few data points or even a single one. The primary goal for this stakeholder is to explain why the data point is classified to the given class (single sample decision understanding). The model creator is building the ML model, has plenty of data, and needs to understand how to improve the model performance (model improvement). The model evaluator has an ML model and a lot of data and needs to explain why all the data had these results (model decision understanding). The adversarial attacker has an ML model, and a lot of data and wants to use explanations to extract (steal) the model (model extraction).

While the stakeholders' needs and requirements can benefit the XAI method selection, how to link these requirements to the evaluation criteria remains unsolved. The main problem is that the stakeholders group description does not sufficiently describe the desired properties of the XAI methods. To overcome this issue, we propose the previously mentioned high-level security tasks for each stakeholder that has identified goals and the XAI method for achieving them. Thus, by evaluating the goal fulfillment, we evaluate the influence the XAI has in the particular task.

3.1.1 Model Decision Understanding

A better understanding of the model is one of the main motivations for XAI methods in domains in which the ML models often serve as support tools and in which the user (human) makes the final decision based on the understanding of the model output.

With the increasing size of the modern ML models, it becomes more challenging to use them locally,

and as a result, access to the model itself is limited. Therefore, we assume a black-box scenario for the model understanding task. Secondly, access to the training dataset and model parameters is not assumed. The core premise of the model understanding task is the following: *The output of the XAI method should be as short as possible while capturing the critical components for the provided ML model output*.

We propose to evaluate how well the explanation captures the merit of the model's decision-making given a sample x by using the explanation to transform x into an adversarial example. Given the trained ML model $f(x) \rightarrow \hat{y}$, the explanation system is defined as $e(f,x,\hat{y}) \rightarrow \mathbf{R}^n$ which given an ML model, data point and label, produces the importance scores for the parts of the data point x. Next, we assume a perturbation mechanism $I(x,R^k) \rightarrow x'$ which modifies the data point based on the given vector of importance scores. Lastly, we assume a collection of testing data points $D_{test} = \{x_1, x_2, ..., x_k\}$.

We measure the average amount of perturbation steps required until the data point is adversarial:

$$\frac{1}{|D|}\sum_{i=1}^{|D|}[\![f(I(x_{i-1}, e(f,x_0,\hat{y_0})) = f(x_0)]\!] \qquad (1)$$

Where x_0 is the original sample from D_{test} being gradually more modified by the perturbation mechanism and $[\![expression]\!]$ represents a logical value of the expression inside the brackets.

3.1.2 Single Sample Decision Understanding

The single sample decision understanding is a special variant of the task described in Section 3.1.1 of model understanding when the size of the $D_{test} = 1$. This task refers to the problem of having only one data point (instead of a whole dataset) and therefore being subject to little information to make a decision. This is implemented by randomly selecting one data point from the dataset and running the XAI only on this data point. The previously proposed selection mechanism for the XAI method is still applicable but is prone to high variance due to the lack of data.

3.1.3 Model Improvement

The analysis of model errors and weak points is another area of ML where XAI methods are often used. It differs significantly from the previous task, as such analysis is commonly performed either by the model developer or auditor that has access to the model and its parameters and architecture as well as the **labeled dataset**. The XAI use for model improvement includes augmenting the training data (Weber et al., 2023; Teso and Kersting, 2019).

For the model modification task, we propose to apply the XAI for the augmentation of the training data by identifying the most influential features. Assuming having labeled data $X = \{(x_1,y,1),...,(x_n,y,n)\}$, model $f : X \rightarrow y$ and an XAI method $E(f(x),x,\hat{y}) \rightarrow e_x$. We split X split into three parts X_{train}, $X_{validation}$, X_{test}, and use X_{train} for the training of the model f.

Then, $E(f(x),x,\hat{y})$ is used for every $x \in X_{validation}$ and corresponding model prediction $\hat{y}$ resulting in set of explanations $\{e^{val}_{x_1}, e^{val}_{x_2}, ..., e^{val}_{x_m}\}$. Each e^{val}_x ranks the features of x according to their importance. The augmented training dataset X'_{train} is created by removing the k features with the lowest average ranking and used for re-training the model using the same hyperparameters.

The relative change in the model performance quantifies the impact of the XAI method on the training process. In our experiments, we use F_1 to measure the model performance.

3.1.4 Model Extraction

While the XAI methods provide insight into the model's internal representation and decisions, they can be used in attacks such as model inversion and model extraction (Yan et al., 2023; Kuppa and Le-Khac, 2021). For the evaluation of the applicability of the explanation systems for model attacks, we propose the task of model extraction guided by the explanation.

We assume the target of the attack to be an arbitrary black-box model $f : X \rightarrow y$ which can be queried by the attacker. The goal of the attacker is to construct a surrogate model $f' : X \rightarrow y$ which mimics the behavior of the target model f. The fulfillment of the task is measured by the agreement between the models w.r.t. to a testing dataset X_{test}. The basic model extraction attack consists of the following steps:

1. Generation of a set of **unlabeled** data $X' = \{x'_1, x'_2, ..., x'_n\}$ using original dataset X of limited size
2. Querying the target model f using the $x' \in X'$ and obtaining the labels $\hat{y}$, resulting in training dataset $D' = \{(x'_1, y_1), (x'_2, y_2), ..., (x'_n, y_n)\}$
3. Training of the surrogate f' using D'
4. Evaluation of agreement between the models f and f'

The role of the XAI in this task lies in the generation of training data X' for the surrogate. The explanator output is used to guide the sampling of the dataset X'.For each of the $x \in X$, we use the explanation method $E(f(x),x,\hat{y}) \rightarrow e_x$ to obtain the explanation e_x.

The form of the explanation can vary, in the case of feature importance methods such as SHAP or LIME, the explanation consists of a vector of real numbers $\mathbf{R}^{\|x\|}$ which identifies the importance score for each of components of x. In the case of the Anchor method, the explanation is a sequence of IF-THEN rules that not only identify the feature for which the rule is applicable but also the exact value.

For each of the $x \in X$, k perturbed data points are added to the dataset X'. The perturbation is as follows:

$$x' = x + \varepsilon$$

where ε represents zero mean random vector:

$$\varepsilon \sim \mathcal{N}(0, e_x)$$

Such perturbation strategy allows for better sampling in parts of the feature space which are identified by the explanation method as important for the model's decision making.

We measure the fulfillment of the model extraction task as the agreement between the target model f and the surrogate f'. In our experiments, we use the F1 score for the evaluation. The primary reason is the imbalance of labels in favor of the 'Benign' which is a common problem for cybersecurity problems.

3.2 Dataset

The CTU-50 dataset (Stratosphere, 2015), containing malicious and benign traffic collected in 10 hosts over 5 days, is used. Various malware samples were used to avoid overfitting to a single malware family. The benign traffic captures real users' activity over five consecutive days.

Each data point in the dataset aggregates one-hour time windows for each 4-tuple (*source IP*, *destination IP*, *destination port*, and *protocol*). This 4-tuple identifies all flows related to a specific service. Features related to the TLS traffic of each 4-tuple are aggregated in numerical values, resulting in a feature vector of 38 features used for model training. In total, there are 36168 Malicious samples (Both normal and background traffic) and 1729 Malicious samples.

For the comparison with AutoXAI, the smaller variant of the dataset is included, in which all categorical features as well as highly correlated features are removed to accommodate the limitations of available AutoXAI implementation.

3.3 Experiment Setup

For the experiment, we use four ML models that are commonly used in cybersecurity applications: Random Forest (RF), Gradient Boosting Tree (GBT), Support Vector Machine (SVM), and Multi-Layer Perceptron (MLP). Due to the model variety, we chose to use model-agnostic XAI methods.LIME [1], SHAP[2], Anchors [3], and RISE are used in all experiments. Moreover, a random or uniform baseline for each experiment is included. The implementation can be found in the project repository [4]

For the AutoXAI comparison, we select *fidelity*, *robustness*, and *conciseness* as target properties and 1,1,0.5 as their respective weights.

4 RESULTS

The results of our experiments can be seen in Table 1 for the model understanding task when the replacement of the features is done by using outlier values. The values for evaluating XAI methods are better when they are smaller, meaning fewer interactions were necessary to create adversarial examples because the features selected were very important.

Table 2 shows the same results but for replacing features using the median value of that feature. This corresponds to the idea of *hiding* the value of the feature among the data in order to decrease its importance for the XAI.

Table 3 shows the results for the adversarial attacker task of extracting the models. Table 4 shows the results of the task of model improvement.

Table 1: **Model understanding task - outliers**. The performance of XAI methods is measured as the smallest amount of input data point perturbation required for the creation of an adversarial sample. The important parts of the input data point (as identified by the explanation) are set to the outlier value -1. Lower is better, with 1 being the optimal value.

	LIME	SHAP	Anchor	RISE	Random
Random Forest	1.505	5.180	9.718	7.535	8.660
Gradient Boosting Tree	24.196	35.047	35.704	26.396	27.486
MLP	8.161	7.174	7.066	13.841	16.719
SVM	1.162	1.052	1.263	1.028	1.330

Table 2: **Model understanding task - median**. The performance of XAI methods is measured as the smallest amount of input data point perturbation required for the creation of an adversarial sample. The important parts of the input data point (as identified by the explanation) are set to the median. Lower is better, with 1 being the optimal value.

	LIME	SHAP	Anchor	RISE	Random
Random Forest	1.302	1.048	1.338	1.006	1.526
Gradient Boosting Tree	1.346	1.004	1.316	1.005	1.574
MLP	2.136	2.156	2.64	2.85	4.024
SVM	1.196	1.332	1.35	1.288	1.407

[1]https://pypi.org/project/lime/

[2]https://pypi.org/project/shap/

[3]https://pypi.org/project/anchor-exp/

[4]https://github.com/stratosphereips/sec-xai

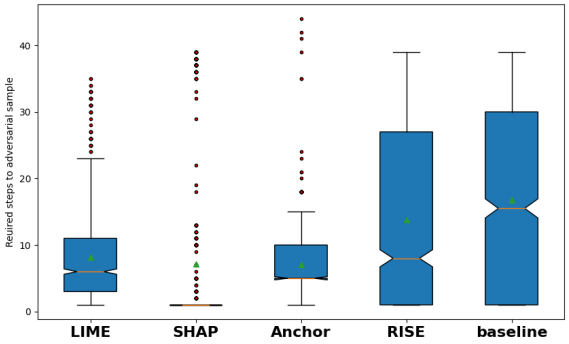

Figure 1: **Model understanding experiment - MLP** Boxplot representation of the results shown the third row of Table 1.

Table 3: **Model extraction**. A comparison of the influence of XAI methods on the surrogate model creation. The model agreement is measured as the F1 score of the output of the target and surrogate models. Higher is better.

target model	surrogate model	LIME	SHAP	Anchor	RISE	Uniform
RF	RF	**0.6857**	0.6479	0.6427	0.5646	0.5544
GBT	RF	0.7053	0.6550	0.5936	0.6106	0.5682
MLP	RF	0.6179	0.6402	0.7588	0.6449	0.6457
SVM	RF	0.7372	0.7418	0.7712	0.0	0.0
RF	GBT	0.6321	0.6501	0.6012	0.6108	0.5038
GBT	GBT	0.6991	0.6882	0.7044	0.8284	0.6413
MLP	GBT	0.6670	0.7358	0.6568	0.6940	0.6250
SVM	GBT	0.7136	0.7378	0.6967	0.3302	-

Table 4: **Model Improvement**. Relative change in the model F1 score after retraining on the dataset augmented by the respective XAI method. Higher is better.

	LIME	SHAP	Anchor	RISE	Random
Random Forest	-0.10%	-0.48%	0.36%	0.62%	-3.11%
Gradient Boosting Tree	-1.17 %	-0.95%	-0.22%	-0.11%	-4.41%
MLP	7.80%	17.52%	-24.90%	2.58	-67.83
SVM	-3.07%	-20.99%	-4.37%	-19.07%	-5.21%

5 ANALYSIS

The results of the experiment show that none of the XAI methods is outperforming the others in all the proposed tasks. A high variance of the results can be observed between the tasks and also between the target ML models.

In the model understanding task, we evaluated the proposed task and metrics in two experiments that differed in the value used for the masking-out of the features identified by the XAI methods.

In Table 1, the masking value is set to a constant, which is **not** present in the original feature values. In Table 2 the feature values were replaced by a median of the corresponding column in the dataset. The result comparison shows, that using the median resulted in lower variance of the method scores.

In the experiment, where features were masked with median, RISE and SHAP explanation resulted in the fastest creation of adversarial samples for the ensemble methods (Random Forrest, Gradient Boosting Trees). At the same time, LIME showed better performance for the Neural network and the SVM. All XAI methods consistently outperformed the baseline mode.

Table 5: Overview of best-performing XAI method per task and target ML model.

Task	Model	Best XAI method
Model understanding (outliers)	RF	LIME
Model understanding (outliers)	GBT	LIME
Model understanding (outliers)	MLP	Anchors
Model understanding (outliers)	SVM	RISE
Model understanding (median)	RF	RISE
Model understanding (median)	GBT	LIME
Model understanding (median)	MLP	LIME
Model understanding (median)	SVM	LIME
Model extraction (RF)	RF	LIME
Model extraction (RF)	GBT	LIME
Model extraction (RF)	MLP	Anchors
Model extraction (RF)	SVM	Anchors
Model extraction (GBT)	RF	SHAP
Model extraction (GBT)	GBT	RISE
Model extraction (GBT)	MLP	SHAP
Model extraction (GBT)	SVM	SHAP
Model improvement	RF	RISE
Model improvement	GBT	RISE
Model improvement	MLP	SHAP
Model improvement	SVM	LIME

Table 6: AutoXAI results for smaller dataset.

target model	Suggested XAI method
RF	LIME
GBT	LIME
MLP	SHAP
SVM	LIME

During the creation of adversarial samples by replacing features with outlier values (Table 1), LIME performed best in all models except for the MLP. Figure 1 shows the boxplot representation of the third row of Table 1. It shows that while, on average LIME required 1 step more than SHAP to create adversarial samples for the MLP, SHAP performed better for most data points.

The results in Table 4 show that in the training data augmentation, LIME was the most consistent XAI method across all of the ML models.

The XAI methods were used to create the surrogate training data for the model extraction task. From the experiment results in Table 3, we can see that LIME performed the best when creating RF surrogates for models of similar types. At the same time, Anchors performed best when the target model architectures were completely different (MLP, SVM), achieving over 70% of agreement with the original model measured by the F1 score. When the GBT was used as a surrogate, SHAP was the most consistent

XAI method for the task.

A comparison with the AutoXAI results shown in 6 shows that while for some model types, both methods suggest the same XAI method, the task separation allows for more precise matching of the explanation system to the desired goal. It is important to note, that further hyperparameter search for the AutoXAI could provide higher granularity in the results.

5.1 Method Limitations

There are some limitations of our proposed method and some future work that would help better evaluate it. Currently, the technique strongly needs feature importance to work, which limits the evaluation of generic XAI methods. Secondly, the evaluation depends on the model and data which makes the results not directly transferable and, the evaluation itself time-consuming. Lastly, the perturbation scheme can not accommodate non-tabular data.

6 CONCLUSION

This paper proposes an approach to evaluating model-agnostic explanation methods focusing on the network security domain.

Our method evaluates the explanation systems in the context of tasks in which the explanations are used. We formulated and evaluated three high-level tasks that represent typical applications of XAI methods in the development and usage of ML models in security, each represented by a triplet of dataset, ML model, and goal. In the experimental evaluation, we compared four model-agnostic XAI methods with a baseline (LIME, SHAP, RISE, Anchors) in each proposed task. The evaluation included four model types (Random forest, Gradient Boosting Trees, SVM, and MLP.

The experimental evaluation showed that the formulated tasks are diverse and that none of the xai methods outperformed the others. That supports the hypothesis of the importance of the relationship between the XAI method and the underlying ML model. In particular, in most of the experiments, LIME showed the best results when the underlying model was a Random Forest classifier, whereas for the Neural networks and SVM, SHAP and Anchors performed better.

We have further shown that the proposed tasks can be used for the evaluation of the XAI methods for network security.

6.1 Future Work

Adapting the method to other XAI is planned to broaden the usability of the proposed evaluation framework. Such adaptation includes support for other data types such as time series or graphs. The second extension includes evaluating a wider variety of security datasets or adaptations to other domains and the incorporation of our evaluation in the AutoXAI framework.

ACKNOWLEDGEMENTS

The authors acknowledge support from the Strategic Support for the Development of Security Research in the Czech Republic 2019–2025 (IMPAKT 1) program, by the Ministry of the Interior of the Czech Republic under No. VJ02010020.

REFERENCES

Agarwal, C., Krishna, S., Saxena, E., Pawelczyk, M., Johnson, N., Puri, I., Zitnik, M., and Lakkaraju, H. (2022). Openxai: Towards a transparent evaluation of model explanations. In Koyejo, S., Mohamed, S., Agarwal, A., Belgrave, D., Cho, K., and Oh, A., editors, *Advances in Neural Information Processing Systems*, volume 35, pages 15784–15799. Curran Associates, Inc.

Aledhari, M., Razzak, R., and Parizi, R. M. (2021). Machine learning for network application security: Empirical evaluation and optimization. *Computers & Electrical Engineering*, 91:107052.

Alvarez-Melis, D. and Jaakkola, T. S. (2018). On the Robustness of Interpretability Methods. arXiv:1806.08049 [cs, stat].

Amit, I., Matherly, J., Hewlett, W., Xu, Z., Meshi, Y., and Weinberger, Y. (2019). Machine Learning in Cyber-Security - Problems, Challenges and Data Sets. *arXiv:1812.07858 [cs, stat]*. arXiv: 1812.07858.

Arrieta, A. B., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., García, S., Gil-López, S., Molina, D., Benjamins, R., Chatila, R., and Herrera, F. (2019). Explainable Artificial Intelligence (XAI): Concepts, Taxonomies, Opportunities and Challenges toward Responsible AI. arXiv:1910.10045 [cs].

Arrieta, A. B., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., García, S., Gil-López, S., Molina, D., Benjamins, R., et al. (2020). Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. *Information fusion*, 58:82–115.

Cugny, R., Aligon, J., Chevalier, M., Roman Jimenez, G., and Teste, O. (2022). Autoxai: A framework to automatically select the most adapted xai solution. In *Proceedings of the 31st ACM International Conference on*

Information & Knowledge Management, CIKM '22, page 315–324, New York, NY, USA. Association for Computing Machinery.

Doshi-Velez, F. and Kim, B. (2017). Towards A Rigorous Science of Interpretable Machine Learning. arXiv:1702.08608 [cs, stat].

Jacobs, A. S., Beltiukov, R., Willinger, W., Ferreira, R. A., Gupta, A., and Granville, L. Z. (2022). AI/ML for Network Security: The Emperor has no Clothes. In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security*, pages 1537–1551, Los Angeles CA USA. ACM.

Jin, W., Fan, J., Gromala, D., Pasquier, P., and Hamarneh, G. (2023). Invisible Users: Uncovering End-Users' Requirements for Explainable AI via Explanation Forms and Goals. arXiv:2302.06609 [cs].

Kim, B., Khanna, R., and Koyejo, O. O. (2016). Examples are not enough, learn to criticize! Criticism for Interpretability. In Lee, D., Sugiyama, M., Luxburg, U., Guyon, I., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc.

Krishna, S., Han, T., Gu, A., Pombra, J., Jabbari, S., Wu, S., and Lakkaraju, H. (2022). The Disagreement Problem in Explainable Machine Learning: A Practitioner's Perspective. arXiv:2202.01602 [cs].

Kuppa, A. and Le-Khac, N.-A. (2021). Adversarial xai methods in cybersecurity. *IEEE transactions on information forensics and security*, 16:4924–4938.

Lundberg, S. and Lee, S.-I. (2017). A Unified Approach to Interpreting Model Predictions. arXiv:1705.07874 [cs, stat].

Molnar, C. (2022). *Interpretable Machine Learning*. Christoph Molnar, 2 edition.

Mothilal, R. K., Mahajan, D., Tan, C., and Sharma, A. (2021). Towards Unifying Feature Attribution and Counterfactual Explanations: Different Means to the Same End. In *Proceedings of the 2021 AAAI/ACM Conference on AI, Ethics, and Society*, pages 652–663. arXiv:2011.04917 [cs].

Nadeem, A., Vos, D., Cao, C., Pajola, L., Dieck, S., Baumgartner, R., and Verwer, S. (2022). SoK: Explainable Machine Learning for Computer Security Applications. arXiv:2208.10605 [cs].

Nguyen, H. D., Do, N. V., Tran, N. P., Pham, X. H., Pham, V. T., and Minutolo, A. (2020). Some criteria of the knowledge representation method for an intelligent problem solver in stem education. *Appl. Comp. Intell. Soft Comput.*, 2020.

Parmar, M. (2021). Xaisec-explainable ai security: An early discussion paper on new multidisciplinary subfield in pursuit of building trust in security of ai systems.

Petsiuk, V., Das, A., and Saenko, K. (2018). Rise: Randomized input sampling for explanation of black-box models.

Ribeiro, M. T., Singh, S., and Guestrin, C. (2016). "Why Should I Trust You?": Explaining the Predictions of Any Classifier. arXiv:1602.04938 [cs, stat].

Ribeiro, M. T., Singh, S., and Guestrin, C. (2018). Anchors: High-Precision Model-Agnostic Explanations. *Proceedings of the AAAI Conference on Artificial Intelligence*, 32(1). Number: 1.

Sacher, D. (2020). Fingerpointing False Positives: How to Better Integrate Continuous Improvement into Security Monitoring. *Digital Threats: Research and Practice*, 1(1):1–7.

Sarker, I. H. (2021). Machine Learning: Algorithms, Real-World Applications and Research Directions. *SN Computer Science*, 2(3):160.

Schwalbe, G. and Finzel, B. (2023). A Comprehensive Taxonomy for Explainable Artificial Intelligence: A Systematic Survey of Surveys on Methods and Concepts. *Data Mining and Knowledge Discovery*. arXiv:2105.07190 [cs].

Sokol, K. and Flach, P. (2020). Explainability fact sheets: A framework for systematic assessment of explainable approaches. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, FAT* '20, page 56–67, New York, NY, USA. Association for Computing Machinery.

Stratosphere (2015). Stratosphere laboratory datasets. Retrieved March 13, 2020, from https://www.stratosphereips.org/datasets-overview.

Teso, S. and Kersting, K. (2019). Explanatory interactive machine learning. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, AIES '19, page 239–245, New York, NY, USA. Association for Computing Machinery.

van der Waa, J., Nieuwburg, E., Cremers, A., and Neerincx, M. (2021). Evaluating XAI: A comparison of rule-based and example-based explanations. *Artificial Intelligence*, 291:103404.

Vigano, L. and Magazzeni, D. (2020). Explainable Security. In *2020 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*, pages 293–300, Genoa, Italy. IEEE.

Wachter, S., Mittelstadt, B., and Russell, C. (2017). Counterfactual Explanations Without Opening the Black Box: Automated Decisions and the GDPR. *SSRN Electronic Journal*.

Warnecke, A., Arp, D., Wressnegger, C., and Rieck, K. (2020). Evaluating Explanation Methods for Deep Learning in Security. In *2020 IEEE European Symposium on Security and Privacy (EuroS&P)*, pages 158–174.

Weber, L., Lapuschkin, S., Binder, A., and Samek, W. (2023). Beyond explaining: Opportunities and challenges of xai-based model improvement. *Information Fusion*, 92:154–176.

Yan, A., Huang, T., Ke, L., Liu, X., Chen, Q., and Dong, C. (2023). Explanation leaks: Explanation-guided model extraction attacks. *Information Sciences*, 632:269–284.

Yeh, C.-K., Hsieh, C.-Y., Suggala, A. S., Inouye, D. I., and Ravikumar, P. (2019). On the (In)fidelity and Sensitivity for Explanations. arXiv:1901.09392 [cs, stat].