Introducing a Multi-Perspective xAI Tool for Better Model Explainability

Marek Pawlicki[1,2], Damian Puchalski[1], Sebastian Szelest[1], Aleksandra Pawlicka[1,3], Rafał Kozik[1,2], and Michał Choraś[1,2]

[1]**ITTI Sp. z o.o., Poznań, Poland**
[2]**Bydgoszcz University of Science and Technology, Bydgoszcz, Poland**
[3]**University of Warsaw**

ABSTRACT

This paper introduces an innovative tool equipped with a multi-perspective, user-friendly dashboard designed to enhance the explainability of AI models, particularly in cybersecurity. By enabling users to select data samples and apply various xAI methods, the tool provides insightful views into the decision-making processes of AI systems. These methods offer diverse perspectives and deepen the understanding of how models derive their conclusions, thus demystifying the "black box" of AI. The tool's architecture facilitates easy integration with existing ML models, making it accessible to users regardless of their technical expertise. This approach promotes transparency and fosters trust in AI applications by aligning decision-making with domain knowledge and mitigating potential biases.

Keywords: machine learning; AI explainability; network intrusion detection

1 INTRODUCTION

In a variety of domains and applications, the accuracy of Artificial Intelligence (AI) algorithms has surpassed many traditional methods of classification and pattern recognition in data streams. This advancement is also beneficial in cyberattack detection, where, despite different attacker motivations (Pawlicka et al., 2021, 2020), AI shows great potential (Kaur et al., 2023; Rafy, 2024; Pawlicki et al., 2023; Xu et al., 2021). The majority of the high-performing AI algorithms work as black-boxes – the data is fed to the algorithm, and the algorithm provides highly accurate output, however with no indications or explanations regarding the basis of these outputs and the factors influencing the decision-making process (Choraś et al., 2020). In many applications, this knowledge is very valuable, both for AI model users, as well as model developers. Therefore, explainability of AI (xAI) and interpretation of black-box algorithms are of crucial importance for the proliferation of AI technologies in the cybersecurity domain, as one of the ways to garner user trust (Barredo Arrieta et al., 2020; Ribeiro et al., 2016; Pawlicki et al., 2024).

This paper introduces an innovative tool featuring a user-friendly dashboard designed to significantly enhance interpretability for end-users. This dashboard allows users to selectively examine data samples classified by the AI model and choose from an array of explanation methods. Each method provides a distinct perspective into the model's decision-making processes, effectively demystifying the 'black box'. This multi-perspective capability, accessible via an intuitive interface, represents a key contribution of this work, offering users a tailored and insightful exploration of AI outputs.

The rest of the paper is structured as follows:

- Section 2 is focused on the description of related work in the context of applying different methods of xAI.
- The goal of section 3 is to introduce our innovative explainer for AI-based NIDS (Network Intrusion Detection).
- Section 4 presents the local and global explainability capabilities provided by the proposed tool.
- Section 5 concludes the paper.

2 RELATED WORK

A number of recent publications show a successful application of explainable AI models used to detect or classify phenomena for cybersecurity purposes. The need for explainability capabilities is emphasized,

taking into account aspects of ensuring the trustworthiness, interpretability, robustness and performance of AI-based detection models at the local level for each sample, and in the global perspective of the entire decision-making process of the AI model (Mane and Rao, 2021; Hariharan et al., 2023; Neupane et al., 2022). The importance of xAI algorithms is unquestionable for cybersecurity operators using black-box models, but also crucial for the performance of the developers of AI tools for security solutions, ensuring AI reliability or fine-tuning the solutions.

The authors of (Abou El Houda et al., 2022) presented the framework providing real-time intrusion detection for Internet of Things (IoT) networks, enhanced by three xAI techniques, i.e., RuleFit, Local Interpretable Model-Agnostic Explanations (LIME), and SHapley Additive exPlanations (SHAP), on top of a DNN-based detection model. The aim of (Warnecke et al., 2020) was to investigate six popular explanation methods and to evaluate their performance in malware detection and vulnerability discovery scenarios. Assessed methods included LIME, SHAP, Gradients and Integrated Gradients (IG), Local Explanation Method using Nonlinear Approximation (LEMNA) and Layer-Wise Relevance Propagation (LRP). The authors investigated those methods introducing such criteria as descriptive accuracy, descriptive sparsity, as well as completeness, efficiency, robustness and stability of explanation. The works conclude that there exist significant differences between the xAI methods performance based on the particular AI-assisted security task. On the other hand, the authors of (Arreche et al., 2024) proposed an end-to-end framework to evaluate xAI methods performance in network intrusion detection tasks, including global and local explanations. The authors proposed and analyzed different metrics to showcase the differences between popular black-box xAI techniques, namely SHAP and LIME.

Additionally, several open-source, multi-method xAI frameworks, libraries and tools aiming at improved explainability of AI models are available. XAITK toolkit is comprised of variety of separate tools developed at different levels of maturity and addressing different xAI tasks, e.g.: After Action Review for AI, Bayesian teaching for xAI, Counterfactual explanations, fault-line image explanations, similarity-based saliency maps, etc. (Hu et al., 2021). The explAIner framework (Spinner et al., 2019), a tool for interactive and explainable machine learning, incorporates over 20 state-of-the-art xAI methods and helps to understand xAI process mapping it into an iterative, three-stage workflow, including model understanding, diagnosis, and refinement. OmniXAI is an open-source Python library designed to address explaining decisions made by AI models to improve analytic, debugging and interpretability capabilities in a variety of tasks and AI applications (Yang et al., 2022). It includes a range of well-known explanation methods, such as feature-attribution/importance explanation (LIME, SHAP, Integrated Gradients, Grad-CAM, L2X), counterfactual explanation (MACE), Partial Dependence Plots (PDP), and model-specific methods (linear and tree models).

3 EXPLAINER FOR AI-BASED NETWORK INTRUSION DETECTION DECISIONS

The tool proposed in this paper has been developed within the AI4CYBER (Trustworthy Artificial Intelligence for Cybersecurity Reinforcement and System Resilience) project, co-funded by the European Commission. The project provides next-generation trustworthy cybersecurity services that leverage AI and Big Data technologies (ai4, 2024; cor, 2024). The aim of the project is to support system developers and operators in effectively managing robustness, resilience, and dynamic response against advanced and AI-powered cyberattacks. The solution presented in this publication is a sub-component developed within the toolkit focusing on the aspects of the AI4CYBER services trustworthiness, namely explainability (presented TRUST4AI.XAI explainer).

The high-level architecture of the solution is presented in Fig. 1. The information flow starts from the user, who having authenticated, can see the dashboard and issue requests. The tool is based on the publish-subscribe architecture, using the bus to move samples around the different microservices. AI Model Integration Interface is implemented to handle communication with different AI models used in the project, and the integration microservice is used to poll the models when the particular samples are being explained. Next, different pre-processing services are used to transform the data, to tailor it to the different xAI algorithms. Then, the actual xAI microservices are available through, and since some of them are computationally demanding, a data storage component is implemented to save both the time and the resources - when the user needs to see the same explanation multiple times.

The AI-based intrusion detection component, which feed the data to the explainer module, has been trained on the LITNET benchmark dataset (Damasevicius et al., 2020). The xAI module allows the user to view the decision making process of the AI-based detector from multiple perspectives, listed as options in the dashboard.

In the proposed TRUST4AI.XAI component providing explainability features for AI4Cyber services several state-of-the art explanation techniques, along with proprietary methods are implemented to demon-

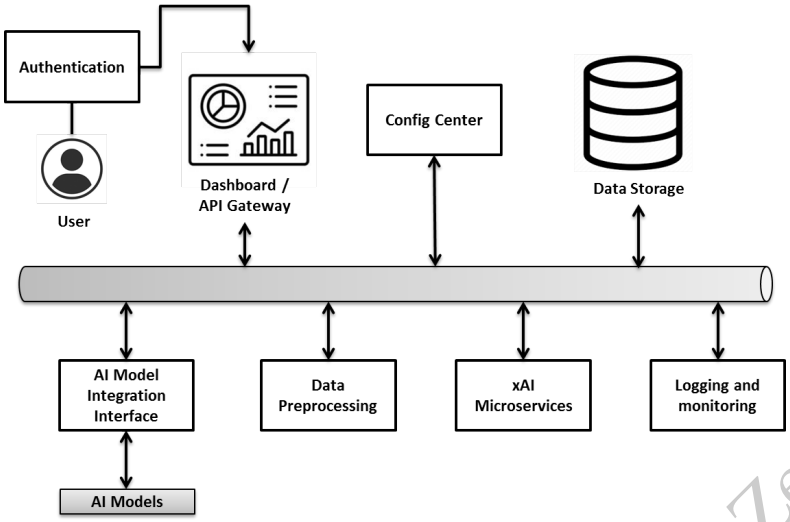

Figure 1. Architecture of the proposed xAI module.

strate different xAI capabilities that can be tailored or customized to different scenarios. There are:

1. Local explainers:
    - Anchors explanations
    - Diverse Counterfactual Explanations (DICE)
    - Local Interpretable Model-Agnostic Explanations (LIME)
    - Shapley Additive Exlanations (SHAP)
    - Explanation based on decision trees
2. Global explainers:
    - Accumulated Local Effects (ALE)
    - Individual Conditional Expectation (ICE)
    - Partial Dependence Plot (PDP)
    - Permutation Feature Importance (PFI)
    - RuleFit method

The summary of the characteristics of each implemented explanation technique and relevant TRUST4AI.XAI dashboard views are presented in the following sub-sections.

4 LOCAL EXPLAINERS

4.1 Anchors explanations

Anchors is a model-agnostic explanation method based on if-then rules, called “anchors”. These rules are specific conditions that, if satisfied, guarantee the same prediction (model output) with a high probability (Ribeiro et al., 2018). Anchors explainer offers insights into a set of factors that are most influential in the AI decision-making process and allows for analysis of the conditions that consistently lead to the same predictions. The authors of this approach demonstrated the usefulness of anchors in different machine learning tasks, namely classification, structured prediction or text generation, for different formats of input data (i.e. tabular, text, images). According to (Ribeiro et al., 2018), the extraction of the partial input data that is sufficient for the classifier to make the prediction, helps to analyze the output of the prediction in an intuitive manner. An example of the AI explanation using this method in the proposed tool is presented in Fig. 2.

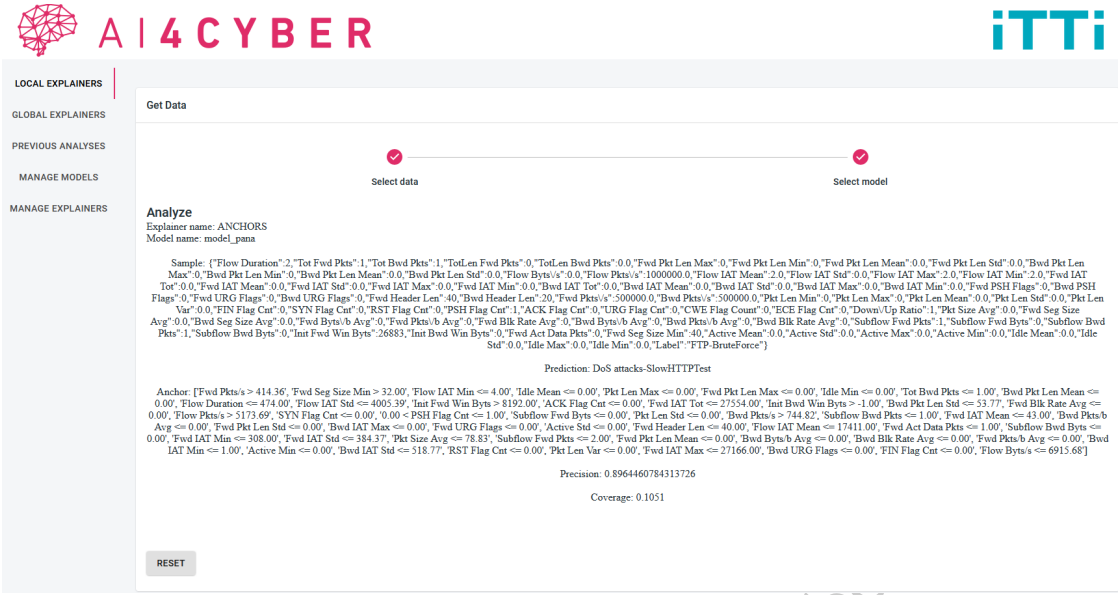

Figure 2. Anchors explanation example.

4.2 Diverse Counterfactual Explanations (DICE)

DICE (Diverse Counterfactual Explanations) is a method based on the generation of counterfactuals – hypothetical scenarios showing what minimal changes are needed to obtain different predictions. In other words, this method provides a set of different "if-then" predictions by perturbing input information, to understand the ML model's decision process and its boundaries (Mothilal et al., 2020). In the proposed explainer module, the user can analyze the original outcome of the model and several feasible counterfactual examples, as can be seen in Fig. 3.

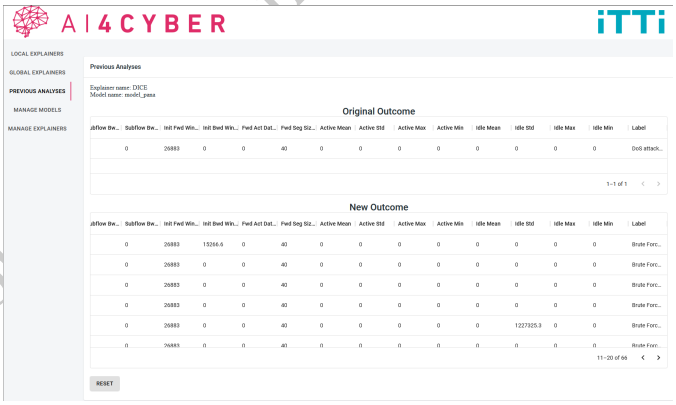

Figure 3. DICE explanation example.

4.3 Local Interpretable Model-Agnostic Explanations (LIME)

LIME (Local Interpretable Model-Agnostic Explanations) is a model-agnostic algorithm that provides explanations based on a generated "local", simpler model that approximates the behaviour of the original model around a specific input instance. More specifically, in the first step, the instance to be explained is selected, and perturbed versions of the instance are generated by randomly masking or adding noise to the features. Then, the weights of the interpretable features are computed to obtain the importance of each feature in the prediction. This is accomplished by training a linear model on the perturbed instances, where the interpretable features are used as input and the predicted probability of the original model is the output (Ribeiro et al., 2016; Molnar et al., 2020). In the proposed explainer, the explanation is visualized by the plot showing feature importance attributed by LIME, as presented in Fig. 4.

4.4 Shapley Additive Explanations (SHAP)

SHAP (SHapley Additive exPlanations) is a model-agnostic machine learning explanation method based on assigning value points to each input feature to explain the prediction. It is done by showing to which

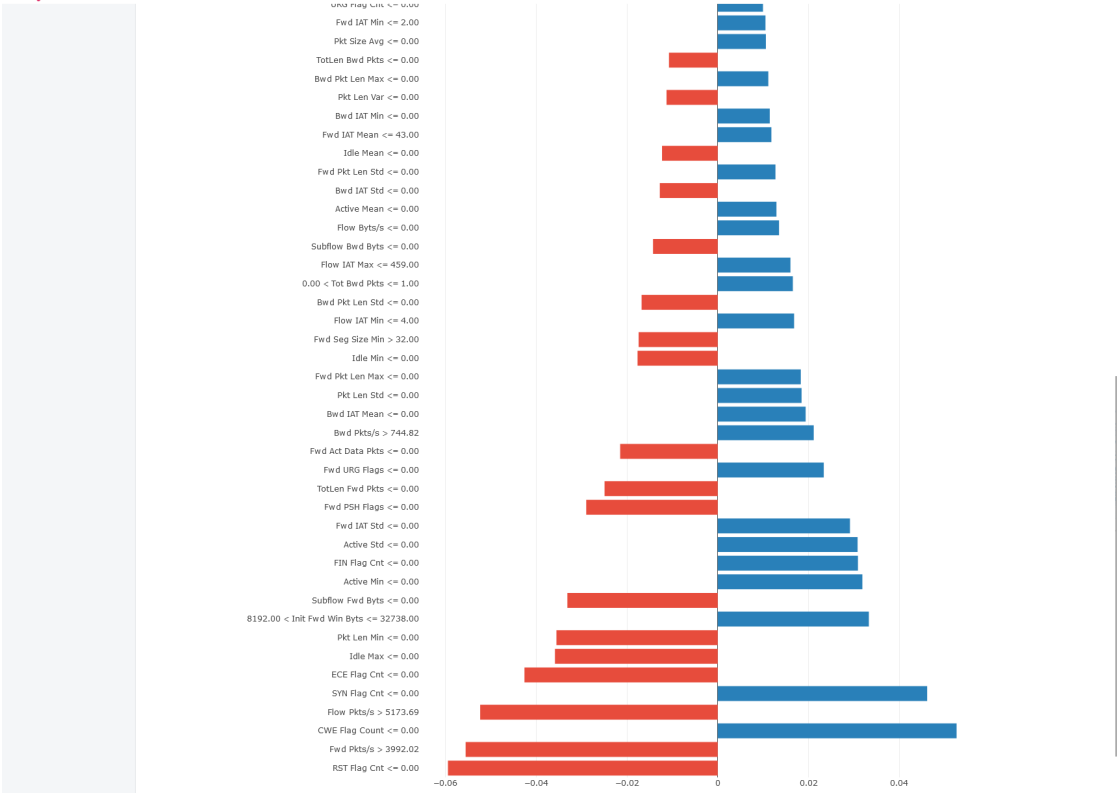

Figure 4. LIME explanation example.

extent a specific feature contributes to the final prediction. The SHAP score represents the difference between expected predictions when the feature exists and when the feature does not exist, calculated on the average for all the possible combinations of features (Molnar et al., 2020; Lundberg and Lee, 2017). Complete interpretation of the prediction is visualized by displaying these values for each feature on a bar or summary plot, as shown in Fig. 5.

4.5 Tree-based explainer

The decision tree method is a model-agnostic technique to provide model explanations by constructing a tree-like model of decisions and their potential repercussions. The algorithm ingests the sample along with the classification performed by the AI model and compares them with pre-trained explanation clusters to find the closest match. Each node in the tree indicates a choice based on a certain data aspect or attribute, and the edges reflect the various consequences of that decision (Quinlan, 1986). The decision tree linked to the closest cluster is used to explain the model prediction (Fig. 6).

5 GLOBAL EXPLAINERS

5.1 Accumulated Local Effects (ALE)

ALE (Accumulated Local Effects) method to visualize and interpret model behaviour has been introduced in (Apley and Zhu, 2020). This algorithm computes the model predictions for data points within intervals of a particular feature. After calculating predictions for different intervals, while keeping other variables fixed, the algorithm calculates differences between adjacent predictions, accumulates the effects and plots them. The visualized output for a given range of a specific feature allows for understanding the influence of this feature in predictions (Fig. 7). In particular, this algorithm can be useful to detect non-linear relationships between variables and the model output.

5.2 Partial Dependence Plot (PDP)

PDP (Partial Dependence Plot) is a technique to visualize the models predicted outcome, based on the feature changes. In general, the approach is an alternative to the ALE technique. However, instead of accumulating local effects, the visualization is provided based on the average of the model predictions over a dataset consisting of selected features in a specific range (Friedman, 2001). PDP helps in understanding

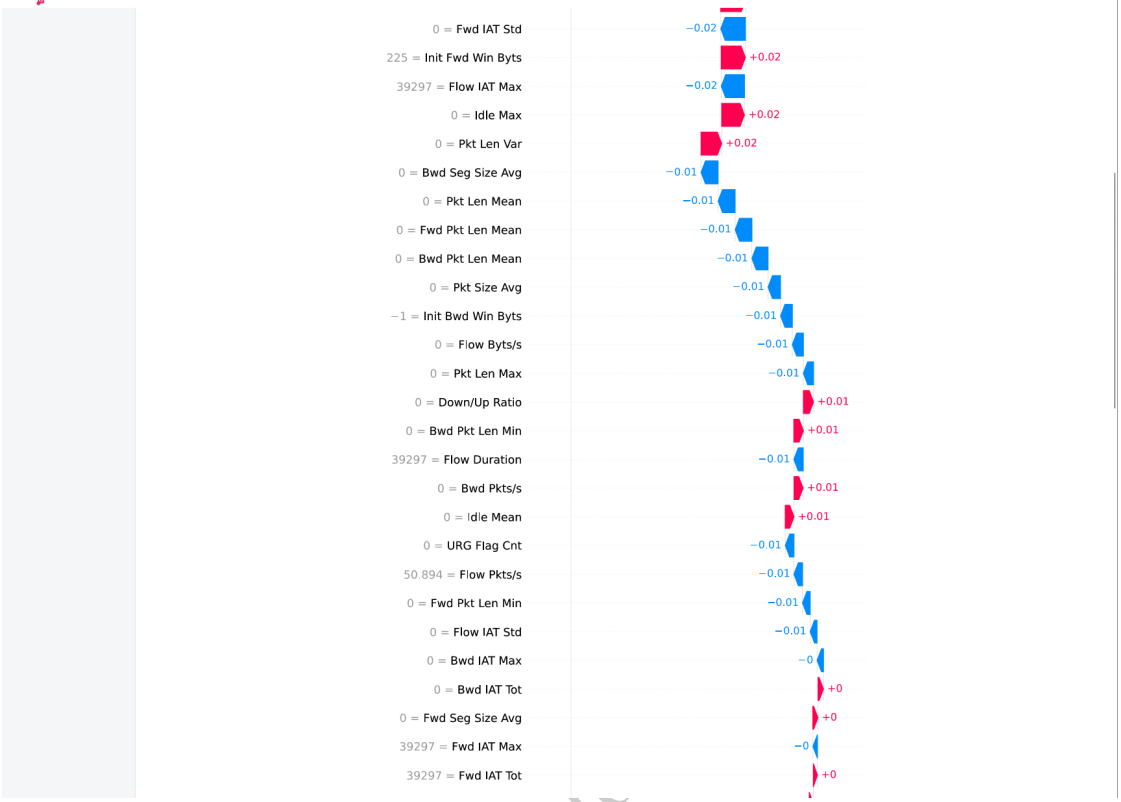

Figure 5. SHAP explanation example.

the global effect of the given feature, ignoring interaction with other features and is useful to interpret complex models by providing a straightforward representation of the feature influence on the predicted output, as presented in Fig. 8.

5.3 Individual Conditional Expectation (ICE)

ICE (Individual Conditional Expectation) plots are another technique used for understanding the relationship between input variables and model predictions, similar to PDP and ALE plots. ICE plots provide a visualization showing the prediction behaviour for each instance individually. This results in plotting one line per instance as shown in Fig. 9, in contrary to one line overall plotted using the PDP method (Goldstein et al., 2013; Molnar et al., 2020).

In general, the PDP provides visualization of the average of the lines calculated and plotted using the ICE method. In the proposed dashboard, there is also an included explainer combining the ICE and PDP plots (Fig. 10, thus providing a more comprehensive view of how selected features influence the model prediction both on average and for varied individual instances. This can be helpful in highlighting of potential variations and anomalies that might be overlooked using only a single method.

5.4 Permutation Feature Importance (PFI)

PFI (Permutation Feature Importance) is a model-agnostic technique helping in model interpretation based on measurements of the model accuracy by shuffling the feature values. In other words, PFI measures the variation of the prediction error applying permutation of the feature values. Based on this approach, a given feature can be evaluated as important to the model's decisions when shuffling its values increases the model error because in this case, the model relied on the feature for the prediction. The model ignores the feature or the future is less important when changes in its output do not impact the prediction error (Breiman, 2001; Molnar et al., 2020).

5.5 RuleFit

RuleFit shown in Fig. 12, is a method in which a set of rules generated from decision trees is combined with linear models. The final model is a linear combination of these rules and original features making predictions based on both. To identify the non-linear relationships in the data, the RuleFit algorithm first constructs a decision tree ensemble, often a random forest. Using a process known as rule extraction, the

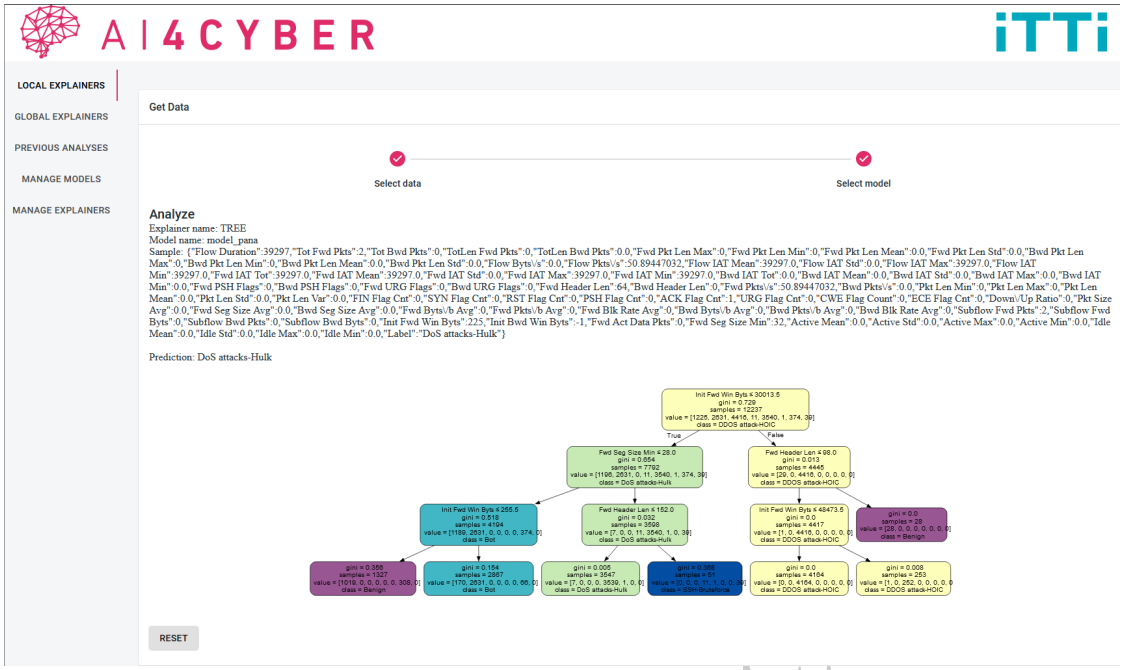

Figure 6. Decision tree explanation example.

decision tree algorithm is then turned into a collection of rules. To develop a hybrid model, the extracted rules are merged with linear models such as linear regression or logistic regression. The linear models are utilized to represent the linear patterns of the data, while the non-linear correlations are captured by the retrieved rules from the decision tree ensemble (Friedman and Popescu, 2008; Molnar et al., 2020). The visualization provides insights not only into the feature importance but also shows which different feature combinations affect the model output.

6 CONCLUSIONS

In this paper, a comprehensive explainability module for AI algorithms, called TRUST4AI.XAI has been proposed. The tool has been developed within the AI4CYBER project to provide AI explainability capabilities. The paper presents 10 different explanation and visualization techniques, namely: Anchors, DICE, LIME, SHAP, Decision Tree-based explanations, ALE, PDP, ICE, PFI and RuleFit, implemented as the backend in the tool presented, and available for the user via the intuitive dashboard. Upon setup of the tool, these methods are made available for anyone, regardless of their level of expertise and without necessitating any coding knowledge. This is achieved by setting up a scalable architecture allowing for

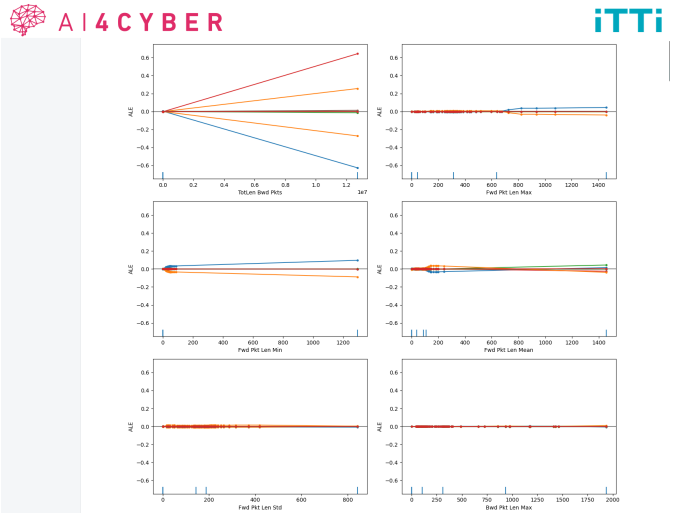

Figure 7. ALE explanation example.

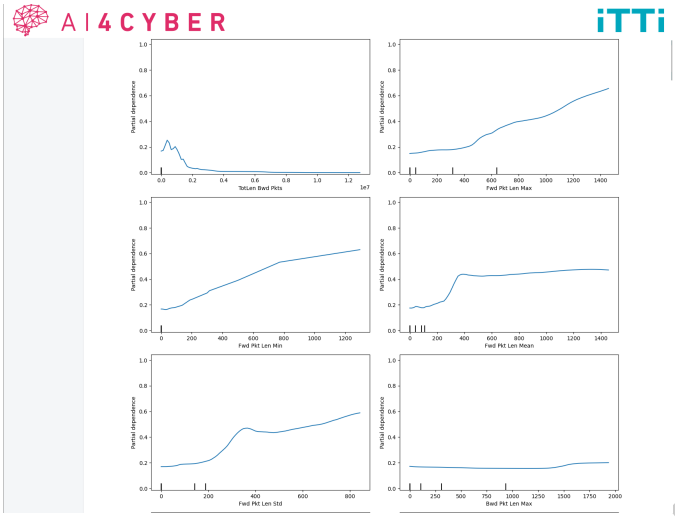

Figure 8. PDP explanation example.

connectivity with various ML models, data preprocessing, and serving the results of the xAI methods to the user in a browser window.

Employing a broad range of different explainability methods on the same model or data sample offers several advantages in comparison to relying only on a single method to interpret the model behaviour. The explanations obtained using different methods provide unique insights into the AI model's decision-making processes, offering a more comprehensive understanding of the underlying logic of the black box.

Parallel use of different explanations can also help to evaluate model robustness, ensuring that the decision-making aligns with domain knowledge. This approach helps to avoid biases that may be observable using one specific method but not another, and facilitates the refining or fine-tuning of the model by allowing a detailed and comprehensive analysis of particular feature importance and its contributions to the model output. The tool is being rolled out to the end users in the AI4Cyber project, and a systematic evaluation of its usability and effectiveness is planned to inform iterative enhancements based on empirical user data.

ACKNOWLEDGMENTS

This research is funded under the Horizon Europe AI4CYBER Project, which has received funding from the European Union's Horizon Europe research and innovation programme under grant agreement No. 101070450

REFERENCES

(2024). Ai4cyber web page at the european commission's cordis portal. `https://cordis.europa.eu/project/id/101070450`. Accessed April 23, 2024.

(2024). Trustworthy artificial intelligence for cybersecurity reinforcement and system resilience (ai4cyber). `https://ai4cyber.eu/`. Accessed April 23, 2024.

Abou El Houda, Z., Brik, B., and Khoukhi, L. (2022). "why should i trust your ids?": An explainable deep learning framework for intrusion detection systems in internet of things networks. *IEEE Open Journal of the Communications Society*, 3:1164–1176.

Apley, D. W. and Zhu, J. (2020). Visualizing the effects of predictor variables in black box supervised learning models. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 82(4):1059–1086.

Arreche, O., Guntur, T. R., Roberts, J. W., and Abdallah, M. (2024). E-xai: Evaluating black-box explainable ai frameworks for network intrusion detection. *IEEE Access*.

Barredo Arrieta, A., Díaz-Rodríguez, N., Del Ser, J., Bennetot, A., Tabik, S., Barbado, A., Garcia, S., Gil-Lopez, S., Molina, D., Benjamins, R., Chatila, R., and Herrera, F. (2020). Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Information Fusion*, 58:82–115.

Breiman, L. (2001). Random Forests. *Machine Learning*, 45(1):5–32.

Chorań, M., Pawlicki, M., Puchalski, D., and Kozik, R. (2020). Machine Learning – The Results Are Not the only Thing that Matters! What About Security, Explainability and Fairness? BT - Computational

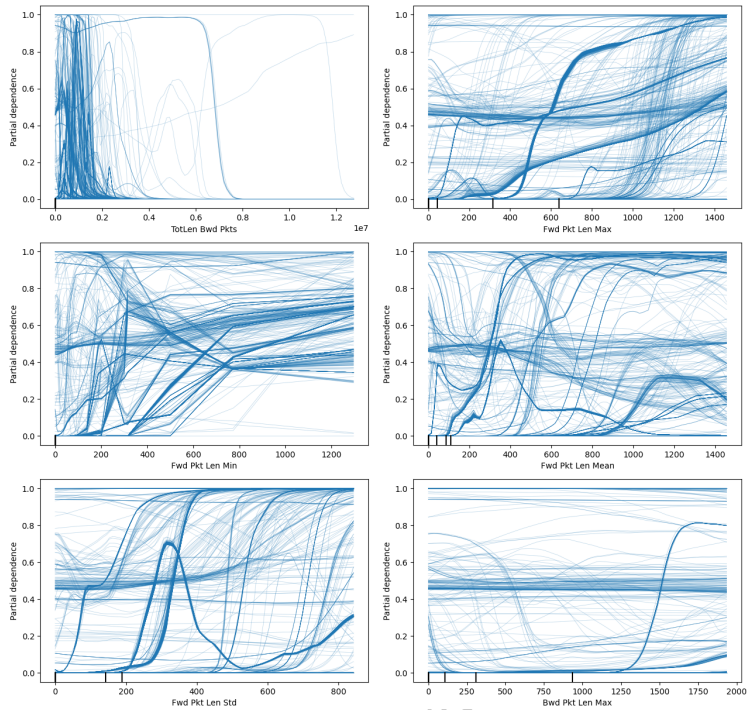

Figure 9. ICE explanation example.

Science – ICCS 2020. In Krzhizhanovskaya, V. V., Závodszky, G., Lees, M. H., Dongarra, J. J., Sloot, P. M. A., Brissos, S., and Teixeira, J., editors, , pages 615–628, Cham. Springer International Publishing.

Damasevicius, R., Venckauskas, A., Grigaliunas, S., Toldinas, J., Morkevicius, N., Aleliunas, T., and Smuikys, P. (2020). Litnet-2020: An annotated real-world network flow dataset for network intrusion detection. *Electronics*, 9(5):800.

Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, 29(5).

Friedman, J. H. and Popescu, B. E. (2008). Predictive learning via rule ensembles.

Goldstein, A., Kapelner, A., Bleich, J., and Pitkin, E. (2013). Peeking Inside the Black Box: Visualizing Statistical Learning with Plots of Individual Conditional Expectation.

Hariharan, S., Rejimol Robinson, R., Prasad, R. R., Thomas, C., and Balakrishnan, N. (2023). Xai for intrusion detection system: comparing explanations based on global and local scope. *Journal of Computer Virology and Hacking Techniques*, 19(2):217–239.

Hu, B., Tunison, P., Vasu, B., Menon, N., Collins, R., and Hoogs, A. (2021). Xaitk: The explainable ai toolkit. *Applied AI Letters*, 2(4):e40.

Kaur, R., Gabrijelčič, D., and Klobučar, T. (2023). Artificial intelligence for cybersecurity: Literature review and future research directions. *Information Fusion*, 97:101804.

Lundberg, S. M. and Lee, S.-I. (2017). A Unified Approach to Interpreting Model Predictions. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Mane, S. and Rao, D. (2021). Explaining network intrusion detection system using explainable ai framework. *arXiv preprint arXiv:2103.07110*.

Molnar, C., Casalicchio, G., and Bischl, B. (2020). Interpretable machine learning–a brief history, state-of-the-art and challenges. In *Joint European conference on machine learning and knowledge discovery in databases*, pages 417–431. Springer.

Mothilal, R. K., Sharma, A., and Tan, C. (2020). Explaining machine learning classifiers through diverse counterfactual explanations. In *Proceedings of the 2020 conference on fairness, accountability, and transparency*, pages 607–617.

Neupane, S., Ables, J., Anderson, W., Mittal, S., Rahimi, S., Banicescu, I., and Seale, M. (2022). Explainable intrusion detection systems (x-ids): A survey of current methods, challenges, and opportunities. *IEEE Access*, 10:112392–112415.

Pawlicka, A., Choraś, M., and Pawlicki, M. (2020). Cyberspace threats: not only hackers and criminals.

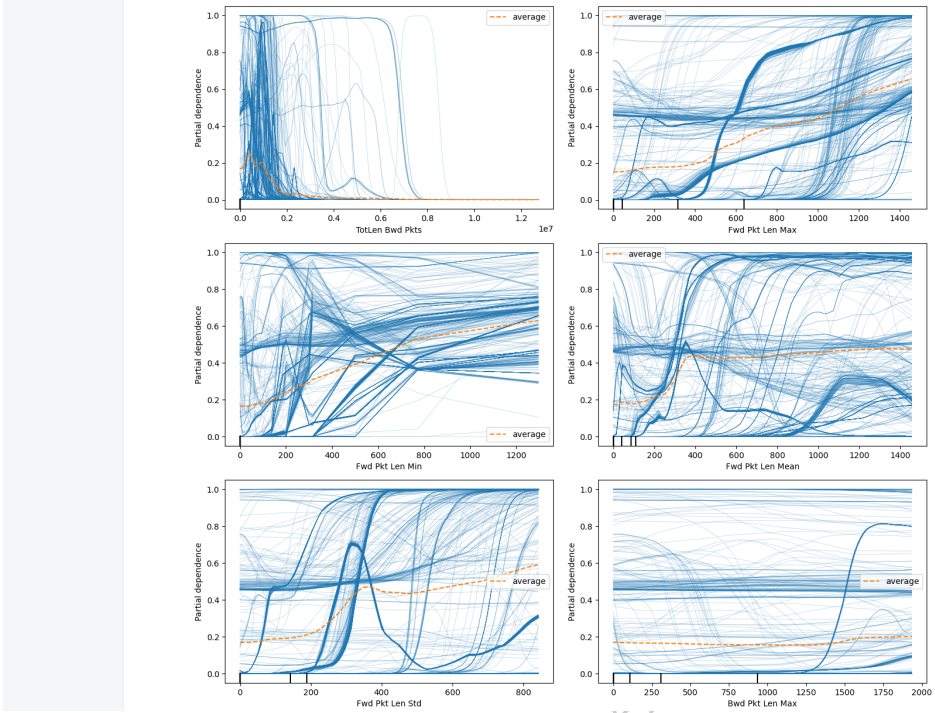

Figure 10. Combination of ICE and PDP explanation – an example.

raising the awareness of selected unusual cyberspace actors-cybersecurity researchers' perspective. In *Proceedings of the 15th International Conference on Availability, Reliability and Security*, pages 1–11.

Pawlicka, A., Choraś, M., and Pawlicki, M. (2021). The stray sheep of cyberspace aka the actors who claim they break the law for the greater good. *Personal and Ubiquitous Computing*, 25(5):843–852.

Pawlicki, M., Pawlicka, A., Kozik, R., and Choraś, M. (2023). The survey and meta-analysis of the attacks, transgressions, countermeasures and security aspects common to the Cloud, Edge and IoT. *Neurocomputing*, 551:126533.

Pawlicki, M., Pawlicka, A., Kozik, R., and Choraś, M. (2024). Advanced insights through systematic analysis: Mapping future research directions and opportunities for xAI in deep learning and artificial intelligence used in cybersecurity. *Neurocomputing*, page 127759.

Quinlan, J. R. (1986). Induction of decision trees. *Machine learning*, 1:81–106.

Rafy, M. F. (2024). *Artificial Intelligence in Cyber Security*.

Ribeiro, M. T., Singh, S., and Guestrin, C. (2016). " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144.

Ribeiro, M. T., Singh, S., and Guestrin, C. (2018). Anchors: High-precision model-agnostic explanations. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32.

Spinner, T., Schlegel, U., Schäfer, H., and El-Assady, M. (2019). explainer: A visual analytics framework for interactive and explainable machine learning. *IEEE transactions on visualization and computer graphics*, 26(1):1064–1074.

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. IEEE.

Xu, Y., Liu, X., Cao, X., Huang, C., Liu, E., Qian, S., Liu, X., Wu, Y., Dong, F., Qiu, C.-W., Qiu, J., Hua, K., Su, W., Wu, J., Xu, H., Han, Y., Fu, C., Yin, Z., Liu, M., Roepman, R., Dietmann, S., Virta, M., Kengara, F., Zhang, Z., Zhang, L., Zhao, T., Dai, J., Yang, J., Lan, L., Luo, M., Liu, Z., An, T., Zhang, B., He, X., Cong, S., Liu, X., Zhang, W., Lewis, J. P., Tiedje, J. M., Wang, Q., An, Z., Wang, F., Zhang, L., Huang, T., Lu, C., Cai, Z., Wang, F., and Zhang, J. (2021). Artificial intelligence: A powerful paradigm for scientific research. *The Innovation*, 2(4):100179.

Yang, W., Le, H., Laud, T., Savarese, S., and Hoi, S. C. (2022). Omnixai: A library for explainable ai. *arXiv preprint arXiv:2206.01612*.

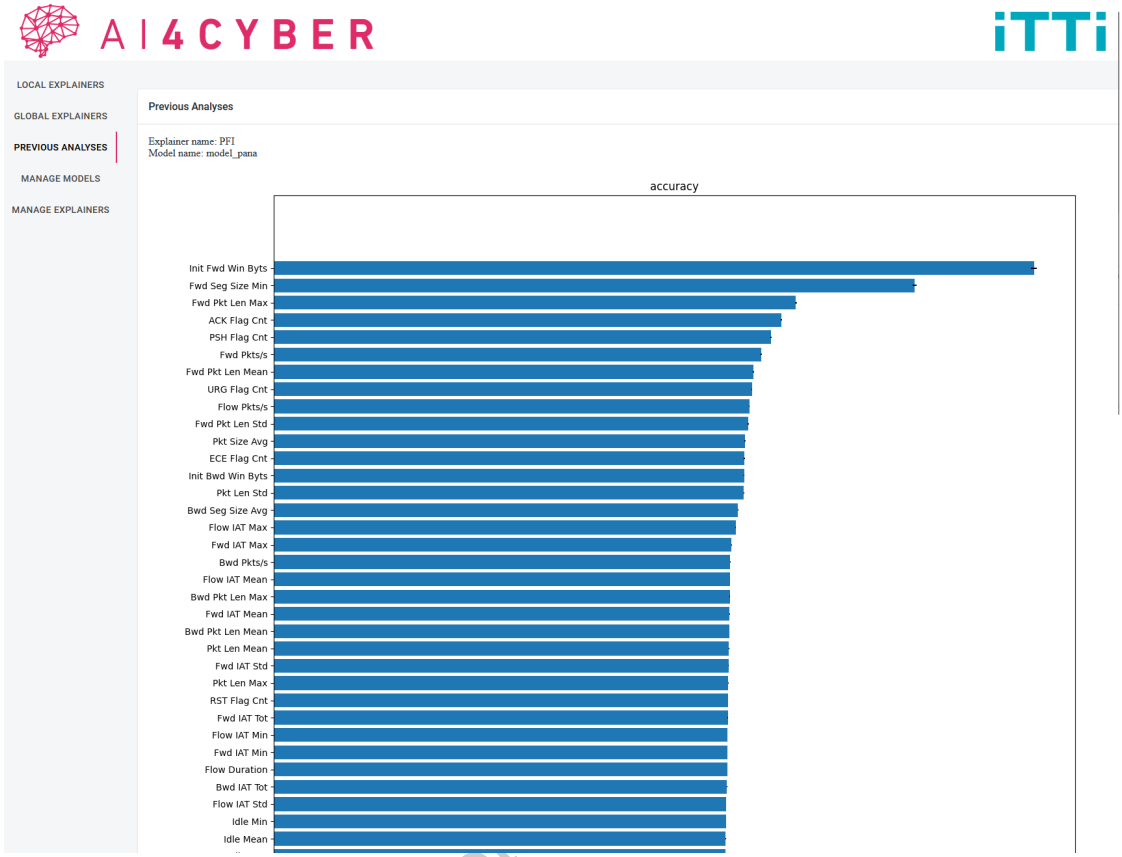

Figure 11. PFI explanation example.

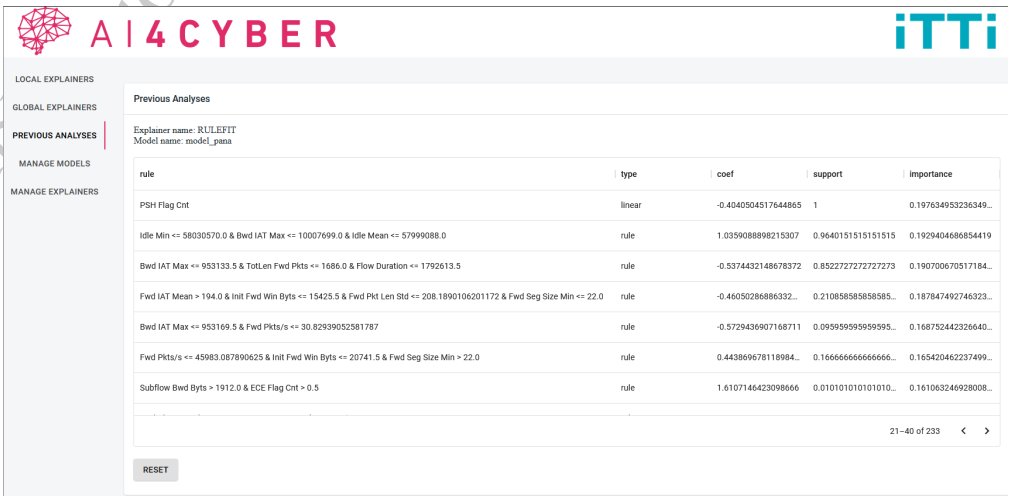

Figure 12. RuleFit explanation example.