Achieving High-fidelity Explanations for Risk Exposition Assessment in the Cybersecurity Domain

1st Albert Calvo
Distributed Artificial Intelligence
i2CAT Foundation
Barcelona, Spain
albert.calvo@i2cat.net

2nd Santiago Escuder
Distributed Artificial Intelligence
i2CAT Foundation
Barcelona, Spain
santiago.escuder@i2cat.net

3rd Josep Escrig
Distributed Artificial Intelligence
i2CAT Foundation
Barcelona, Spain
josep.escrig@i2cat.net

4th Xavier Marrugat
Cybersecurity
i2CAT Foundation
Barcelona, Spain
xavier.marrugat@i2cat.net

5th Nil Ortiz
Cybersecurity
i2CAT Foundation
Barcelona, Spain
nil.ortiz@i2cat.net

6th Jordi Guijarro
Cybersecurity
i2CAT Foundation
Barcelona, Spain
jordi.guijarro@i2cat.net

***Abstract*—Understanding AI-driven systems has become fundamental, particularly when these systems are employed for critical decision-making, as is the case in the field of cybersecurity. In this regard, explainability has been extensively advocated as a cornerstone to comprehend the model, thereby enhancing trust and accountability in data-driven systems. Through the successful use-case of a risk exposure assessment framework which aims to proactively reduce an organization's attack surface, we propose an explainable proxy which is founded on the generation of systematic evaluations of explanations. The proposed framework offers a swift and dependable method for assessing explanations specifically tailored for the cybersecurity domain.**

***Index Terms*—Cybersecurity, Artificial Intelligence, Behavioral Modelling, Explainability**

I. Introduction

Undoubtedly, Artificial Intelligence has become an indispensable driver for almost all contemporary cybersecurity appliances. Its significance is paramount across a diverse array of applications, spanning from network and information security solutions such as Firewalls, Network Access Control (NAC) or Intrusion detection and prevention systems (IDPSs) to the realm of End-user Behavior frameworks – formerly recognized as User and Entity Behavior Analytics (UEBA). Concretely, Artificial Intelligence (AI) has paved the way augmenting the detection, prevention and response capabilities of cybersecurity frameworks by providing out-of-the-box efficient analysis but also enabling security engineers to proactively optimise the daily operations of Security Operation Centers (SOC) whom are responsible of coordinating the cybersecurity technologies and operations of an organization.

However, as responsibilities are increasingly delegated to AI-driven systems, which have increasingly become the preferred option within SOCs worldwide, an issue of trust and accountability in these systems arises when operators cannot comprehend the analytical process and their outcomes. For example, if a certain AI-driven system does not offer adequate mechanisms for comprehending and justifying automatic decision-making, technicians might be hesitant to delegate responsibilities to such systems. This issue is also evident from a compliance perspective. The risk-based regulation, the Artificial Intelligence Act (AIA), proposed by the European Commission regulates AI applications according to different risk levels, with a special focus on providing the attributes of transparency, robustness, and resilience, with particular attention to high-risk use cases [1].

In the last decade, the adoption of AI systems, especially those employing Machine Learning (ML) has experienced exponential growth, thanks to the surge of novel algorithmic techniques and hardware capabilities. In detail, ML is the field of Artificial Intelligence centered on the development of algorithms capable of extracting value from data. These advancements have enabled technicians to analyse large data flows using cost-effective methods, e.g., Gradient Boosting or Deep Learning algorithms. Paradoxically, these algorithms do not always inherently offer a mechanism for transparency, which is crucial for understanding what has the algorithm learned and for providing an explanation for each prediction. To this end, the research area of Explainable Artificial Intelligence (XAI) relies on proposing interpretable methods, but also on utilizing proxies when the preferred algorithm lacks inherent explainability [2].

Our proposed data-driven framework for enterprise security has been successful for computing risk exposition. Currently in production, it offers an out-of-the-box ML-based system that calculates the exposure of each entity within the infrastructure to a specific threat [3]. To achieve this, our tool utilizes different phishing campaigns and analyzes incidental information, enabling it to learn the behavior of users who have been exposed through a supervised approach. The framework propose mitigation and countermeasures with the objective of proactively reduce the attack surface of the entities. In terms of usability, the AI-driven system is restricted to provide a user score (exposition metric) for each entity without providing any augmentation, allowing the stakeholder to justify and better comprehend the outcome of the system.

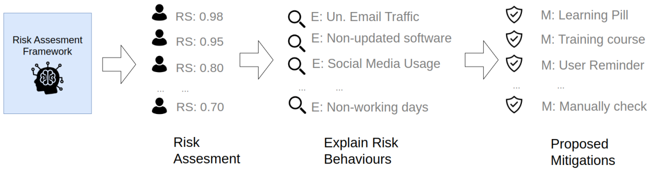


Fig. 1. The risk assessment framework generates a risk exposure score, which is further enriched through the utilization of an explainability proxy. This augmentation facilitates the elucidation of risk behavioral patterns within entities, thereby enabling automated mitigation recommendations for the Security Operations Center (SOC) team.

In this article, the vision and desiderata of a usable explainability proxy in the cybersecurity domain are explained. Specifically, this explainability proxy has been tailored for successful integration into our risk assessment framework. Furthermore, we explore how stakeholders, particularly the Security Operations Center (SOC) team, can derive benefits from these techniques. We detail the explainability procedure implemented within the depicted framework, which empowers us to enhance the exposition metric proposed to stakeholders with explanations. The proposed proxy has been meticulously designed, taking into account the specific requirements of the domain. This design results in high-fidelity explanations that not only provide a reliable justification for automated decisions and countermeasures. In summary, the main contributions of this work are as follows:

- *Usage of evaluation metrics*: We propose to leverage the current advances in the field of explainability by employing evaluation metrics for the systematic assessment of explanations.
- *Organic interface*: In contrast to other proposed AI-driven approaches that build interfaces using out-of-the-box explanations, our proposal leverages evaluation metrics to generate more organic explanations in the cybersecurity domain.
- *Real use case verification*: The proposed explainability proxy is tested within a real-life scenario, involving cybersecurity experts who assess the viability of our methodology.

The remainder of this article is structured in the following manner: Section II provides a comprehensive analysis of the state of the art in explainability and related work within the cybersecurity domain. Section III provides a detailed introduction to our risk analysis framework, and Section IV introduces the proposed explainability proxy. Finally, Section V concludes the paper with a discussion and outlines directions for future research.

II. Explainable Artificial Intelligence (XAI) in the Cybersecurity domain

Explainable Artificial Intelligence (XAI) is formerly described as the science of comprehending what a model did or might have done [4] [5]. To comprehend machine learning models and provide explanations, different XAI approaches are proposed. These approaches are categorized based on whether the models inherently provide a mechanism for explanation (Interpretable-by-design models) or whether Model-Agnostic methods offer an explanation even when the model lacks an inherent approach to be explained [6] [7]. In the cybersecurity domain, comprehending the outputs of machine learning models is foundational. Cybersecurity analysts require well-justified information to make informed decisions, especially considering the critical nature of the information within the SOC, which can potentially impact human lives or result in substantial financial losses.

While various methodologies for explainability (see section II-A) and domain specific use-cases exist in the literature (see section II-B), a common framework to systematically asses explanations has yet to be established. In the technical report proposed in [8], the authors delve into the desiderata of explanations within this domain following a meticulous analysis of the technician requirements. The primary desiderata in this context revolve around the necessity to incorporate temporality and abnormality awareness. Explanations should have the capability to capture the temporal aspect, allowing for the effective analysis of threat actor behavior and various adversarial tactics over time. Furthermore, the explanations provided to end-users should exhibit human-awareness, offering a quick and user-friendly mechanism for obtaining results from AI-driven systems. Lastly, there is a need for quantifying explanations to comprehend their quality and ascertain levels of uncertainty.

A. Methods for XAI

The explainable methods are commonly categorized into intrinsically interpretable models and post-hoc interpretation methods. The distinction between these categories centers around when the explanation is achieved. Intrinsically interpretable models builds the interpretations during the learning stage, whereas post-hoc interpretable models generate explanations after the training process has been completed.

- Intrinsic explainable methods are characterized by the inherent capacity of the algorithms to provide an explanation. A prominent example of such a method is the decision tree algorithm, whose tree-based architecture allows the extraction of explanations in form of rules. Another example of intrinsic methods are the statistic analysis, which employ visual techniques to understand the learning stage [9] [10].
- Post-hoc methods are model-agnostic approaches that employ proxy techniques to approximate both the learning stage and the predictions of the machine learning model, thereby producing an explanation. See, for instance, [11] [12] or [13]. The main types of post-hoc methods are the *visualization methods*, which aim to generate visual insights of the learned model; *knowledge extraction methods*, which extract systematic information from the model; *influence methods*, which estimate the

importance of the features; and *example-based explanations*, which explain the model by choosing particular samples of the dataset and their corresponding outputs [14]. *(1) Visualization methods* are based on graphical methods to interpret the models. For instance, the Partial Dependence Plot (PDP) is a visualization method that calculates the marginal effect between the target and the features in supervised learning models. Another popular visualization technique is surrogate models. These models are based on a proxy that mimics the output of the complex model and is used to comprehend predictions. LIME and SHAP are two popular surrogate techniques that offer both local and global explanation. *(2) Knowledge extraction methods* are target to understand the internal structure of complex models allowing them to provide insights into the models as explanations. The main disadvantage of this type of method is its dependency on the model studied. Rule extraction is a common approach in Deep Neural models where the network is decomposed into multiple decision trees, and the concatenation of rules is used as an explanation. *(3) Influence methods* modify the input instances of the models and observe the effect on the output. For instance, feature importance measures the impact of each feature on the predictions of a complex ML model by perturbing the values of the features and observing the prediction error. *(4) Example-based* methods select representative instances from the dataset to create explanations.

B. XAI in the Cybersecurity Domain

Explanations in the cybersecurity domain are gaining interest since they provide the desired transparency capabilities to cybersecurity appliances using Artificial Intelligence as the core. In the recent years, the number of surveys within Cybersecurity and Explainability has been extensive, see for instance the successful surveys [15] and [16]. Numerous articles employ post-hoc methods such as LIME and SHAP to explain the decisions made by models in the cybersecurity domain. For example, in the intrusion detection field, [17] uses SHAP values to create both local explanations, pertaining to individual samples, and global explanations, providing insights into the model's overall behavior. [18] applies it to for DNS over HTTP (DoH) protocol intrusion detection, while [19] uses SHAP values to compare models trained with CICFlow and NetFlow features. Similarly, [20] uses SHAP to explain Malicious URL classification and Android Malware detection. [21] uses both LIME and SHAP to explain the detection of cryptomining in container clouds. Finally, [22] utilizes a modified version of SHAP values called Shapley–Lorenz for cyberrisk management in order to enhance explainability capabilities.

Furthermore, there are other initiatives that use post-hoc analysis to provide explanations. For instance, [23] has developed TRUST XAI, a model-agnostic explainability tool that has been tested in an IoT environment with both benign and malicious traffic. Additionally, [24] introduces LEMNA, a tool that creates a simpler and interpretable model based on a complex model. This tool was tested in the domain of malware classification and binary reverse engineering. Finally, [25] developed a comprehensive framework that not only included a XAI module but also a data cleaning and an evaluation module that collects feedback from cybersecurity analysts.

III. Our Framework

Our data-driven framework has achieved success in the analysis of a large number of entities from a national public Spanish university. This institution has nearly three thousand nominal users, including research and academic personnel, and produces a flow of 250 GB of logs per day. The objective of the framework is to identify users within the infrastructure at high risk of exposure to potential threats by determining their exposition risk. Despite other data-driven approaches that are based on detection (see for instance [26], [27] or [28]), the objective of the framework is prevention, allowing to determine which users have a behavior which exposes them to a specific threat allowing the SOC operator to take preventive actions and proposing orientated countermeasures to the users.

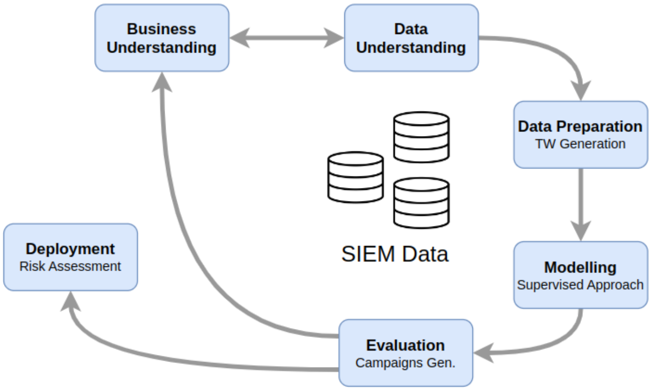


Fig. 2. The risk exposition analysis framework is built following a CRISP-DM methodology

The proposed framework follows the CRISP-DM architecture, which delineates the fundamental stages of common data-driven applications. The initial two steps, Business Understanding and Data Understating, involve an internal assessment of stakeholder requirements and a comprehensive understanding of the data. Concerning to these initial stages, the framework receives data collected from the corporate Security Information and Event Management (SIEM) system. This data source includes raw multi-modal application logs from different entities, including information from DNS, HTTP, SSL, and SMTP application logs. The next step is Data Preparation. Towards the end of the analysis, time windows (TW) are defined to capture the behavior of a user at a specific time interval T_1 - T_0 (see section III-A). The subsequent steps, namely Modelling and Evaluation stages (see sections III-B and III-C), focus on generating a machine learning model. In particular, a binary classification approach is used, utilizing labels from different phishing campaigns. Finally, the model is deployed (see III-D), where the framework facilitates ranking high-risk users. This enables stakeholders to implement appropriate countermeasures and long-term defensive strategies.

A. Data Preparation - TW Generation

To manage the vast volume of raw multi-modal application logs, the data is structured into users and their respective entities. A user, denoted as u, is defined as any individual using the university network infrastructure and possessing an email identity. For the scope of this project, only Administrative and Personnel staff are considered (Eq. 1).

$$U = \{u_1, u_2, u_3, \ldots, u_n\} \tag{1}$$

An entity e refers to any device connected to the network using an IP address, such as workstations, smartphones, printers, or routers that have been discovered in the network. For our case, we consider only workstations and smartphones, as data from the rest of the devices is unavailable (Eq. 2).

$$E_i = \{e_{i1}, e_{i2}, \ldots, e_{im_i}\} \tag{2}$$

Where m_i is the number of entities for user u_i. In order to extract useful information for each entity, we group their application logs into time windows of 10 minutes and calculate a set of statistical features based on ratios and percentages for each time window (Eq. 3).

$$TW_{ij} = \{tw_{ij1}, tw_{ij2}, \ldots, tw_{ijk_i}\} \tag{3}$$

In detail, the features computed for each TW are categorized into five different groups: *DNS Features* are computed with DNS logs generated when using DNS servers. For instance, when an entity wants to access to a website, it first has to resolve the domain name to an IP through a DNS query. *HTTP Features* are computed with HTTP logs. These logs mainly originate from activities related to web-based applications. *SSL Features* are computed with SSL logs generated when using encrypted SSL connections, containing information about the encryption process. *SMTP Features* are computed with SMTP logs. These logs are generated when using the SMTP protocol that is related to emails. *Time-related features*. These features are related to the timing of the logs, such as whether the logs were generated during working hours or not.

B. Modelling - Supervised Approach

The data used to train our model consists of the various time windows that have been calculated. Two distinct models are created for smartphones and workstations, although it could be extended to include other entities in the taxonomy in the future. The objective of these models is to perform binary classification for each time window, allowing the prediction of production data either ”risk” or ”risk-free” behaviors in (Eq. 4).

$$RiskModel = \begin{cases} f_{workstation} : X_{workstation} \rightarrow Y_{workstation} \\ f_{smartphone} : X_{smartphone} \rightarrow Y_{smartphone} \end{cases} \tag{4}$$

Where f refers to the classification model used, and represents X the time window with binary labels referenced as Y. Following a no-free-lunch theorem, our modelling step involves the testing of different Machine Learning models. For instance, the models have been evaluated using the Decision Tree algorithm (DT) [29]. The DT is an interpretable classification algorithm, thus allowing the construction of explanations by analyzing the decision path. Other classification models taken into account include Support Vector Machines (SVM), Gradient Boosting algorithm and Random Forest (RF) [30] [31]. SVM finds a hyperplane that better classifies the different samples. XGBoost is a highly popular ML algorithm across the community for its versatility in tackling classification and regression problems; the algorithm is grounded in gradient boosting tree models [32]. Finally, the last algorithm taken into account is the Random Forest classification algorithm, which creates an ensemble model by combining the output of several decision trees to produce the final output.

C. Evaluation - Phishing Campaigns

To train the individual supervised classification models for the Risk Model, we require distinct labeled datasets. For this purpose, we executed three different simulated email phishing campaigns involving a portion of the university staff. The phishing campaigns involved the creation of various fake emails embedded with tracking elements, which enabling us to monitor user interactions with the emails. This interaction data was then used to label users as compromised or not based on their engagement with the phishing email. In detail, a group of cybersecurity experts elaborated the emails using open-source intelligence tools (OSINT) and publicly available information about the target population to avoid design bias. The first two phishing campaigns aimed at gaining access. The emails asked the user to enter a third-party services, exposing their credentials. Campaign I consisted of requesting login information from a popular e-commerce account, and Campaign II requested login information from their university account. Campaign III focused on malware. The email had a spreadsheet attached with an obfuscated macro. Table I the results of the campaigns can be found. For this article, only the Campaign II is used as it is the most successful campaign. From Campaign II, the *Open* and *Click* users are considered as risk-free users. Whilst the *Engagement* users are considered in risk, the rest of the phishing campaign population is not used in the training of the model, as we do not know how they interacted with the email. Once the labels are extracted, the different entities related to each user and their time windows are labeled, creating a realistic dataset labeled with phishing information.

Once the dataset is created, it is split into 20% validation, 20% test, and 60% train. With the training dataset, four

TABLE I
RESULTS OF PHISHING CAMPAIGNS I-III. THE OBJECTIVE OF CAMPAIGN 1-2 IS GAIN ACCESS. WHILE THE OBJECTIVE OF THE LAST CAMPAIGN IS TO RUN THIRD-PARTY EXECUTABLE.

Campaign-ID	Population	Open	Click	Engagement	Hit-rate
Campaign I	578	156	77	18	3%
Campaign II	377	87	47	67	17%
Campaign III	410	124	-	15	3%

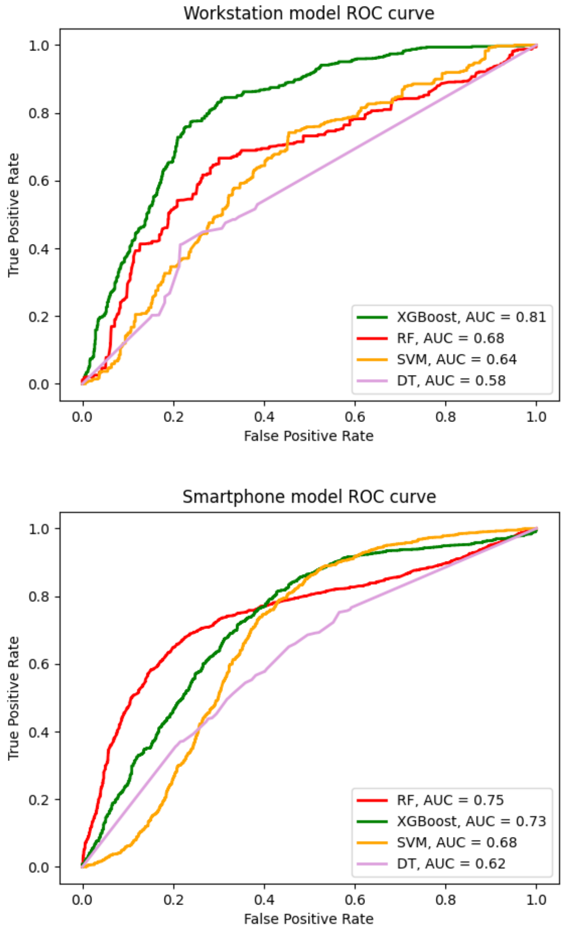


Fig. 3. ROC curves for workstation and smartphone models.

different models are trained using DT, RF, SVM, and Gradient Boosting (XGBoost) algorithms for both smartphones and workstations. A grid search approach has been used to find the best parameters for the models. In Fig. 3, the four different Receiver Operating Characteristic curves (ROC curves) of the different models are shown. ROC curves are a very useful visual tool for binary classification to asses the performance of a model. ROC curves plot the True Positive Rate (TPR) (Eq.5) against the False Positive Rate (FPR), allowing to visually get an insight of how well each model distinguishes between the two classes (Eq.6).

$$TPR = \frac{TP}{TP + FN} \tag{5}$$

$$FPR = \frac{FP}{FP + TN} \tag{6}$$

From the ROC curves, we can calculate the Area Under the Curve (AUC) as a metric for assessing model performance. A higher AUC indicates better discriminative ability of the model. In this project, AUC is used for model selection. Among the models evaluated, the XGBoost algorithm stands out with the highest AUC, especially in the workstation model. In the smartphone models, it is the second-best performer, very closely to Random Forest model (RF). Thus, XGBoost is selected as the primary model. As observed earlier, while Decision Tree (DT) models are transparent and inherently interpretable (white boxes), they lack complexity compared to XGBoost, Support Vector Machine (SVM), or Random Forest (RF) models. Consequently, DT models achieved the lowest AUC among all the proposed models.

D. Deployment - Risk Assessment

To assess the risk of a user, production data is utilized, comprising the application logs of users who were not part of the phishing campaigns. This production data is preprocessed into time windows (TW), and the previously defined and trained Risk Model is applied to compute the predicted accuracy. From now on, the models used are the XGBoost models, represented as $f_{workstation}$ and the $f_{smartphone}$. Both models are designed to perform binary classification of TW, meaning that we can assess if a user is at risk in a certain TW. Moreover, the models also provides the estimated probability of belonging to each class. In this case, as it is binary classification, it returns the probability of the time windows to belong to the class $risk$. The risk exposition of the user can vary depending on the time window. To analyze a user's risk, several consecutive TW can be grouped. Given a period of time with a set of consecutive TW Eq. (7) is applied.

$$\text{score} = \max_{i=1}^{n} \left(\max_{j=1}^{m_i} f_k(e_{ij}) \right) \tag{7}$$

Which means that the $score$ value for a user is the maximum predicted probability of all the TW within a given period of time of the user.

IV. XAI POWERED FRAMEWORK

To enhance the explainability capabilities of our risk assessment framework, we propose a two-step methodology aimed at constructing an automated procedure for assessing explanations. The first step consists of manually identifying behaviors within local explanations from a validation set. This process aims to establish a knowledge database of ground truth explanations employing Shapley values and mapping the explanations to concrete mitigations and countermeasures, as outlined in Section IV-A. The second stage of the explainability proxy involves the systematic assessment of production data. This assessment utilizes faithfulness metrics that enable a comparison between the production data and the previous ground truth explanations, resulting in a score with the alignments that enhances the transparency of the final risk score delivered to stakeholders (refer to Section IV-B).

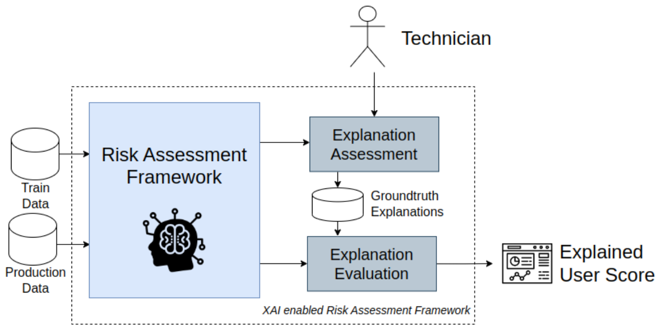


Fig. 4. The proposed explainability proxy comprises a two-stage approach. In the initial stage, diverse ground-truth explanations are labeled to construct a knowledge database. The second stage involves the systematic assessment of explanations using faithfulness metrics.

A. *Explanation Assessment*

In order to explain the predictions made by both Workstation and Smartphone binary classification models, SHAP is used. Specifically, the algorithm enables providing a local explanation, based on determining the most contributing features, for the different TW of a user included in the score. The SHAP (SHapley Additive exPlanations) algorithm draws upon the principles of Shapley values, which stem from the realm of game theory. Shapley values serve as a foundational concept in computing equitable distributions of rewards among participants in a cooperative game. SHAP adapts this concept to attribute contributions of each feature to a specific prediction [33] [6]. In detail, the SHAP method, given a model f and an input x, approximates the output to an explanation model g:

$$f(x) = g(x') = \phi_0 + \sum_{j=1}^{M} \phi_j z'_j \tag{8}$$

Where g represents the explanation model (the additive feature attribution method), M represents the number of input features, and ϕ is the Shapley estimation representing the contribution of each feature. Specifically, the estimation is based on computing N models for all feature subsets and assigning an importance value of including the feature in the final model prediction.

Figure 6 includes diverse explanations collected from the phishing campaigns. The process of labeling the explanations has been performed by plotting the local explanations (top k-Shapley values), and a team of cybersecurity engineers manually asseses the explanations occupying the top positions in the score. Furthermore, Table II contains a comprehensive list of the features that appear in the preceding figure. As follows, the manual assessment for explanations A1-A4 generated using the *Model Workstation* and the *Phishing Campaign II* is included.

- Explanation A1 (predicted probability - 0.72%): This explanation contains two interesting features in the top contributing features. In detail, the $dns\_recursion\_desired\_ratio$ feature might indicate that the entity is making a query requiring a DNS resolver to perform recursive resolutions by querying authoritative DNS servers. On the other hand, the $dns\_qtype\_obsolete\_ratio$ refers that the entity is performing DNS query types that are no longer active or deprecated due to security concerns or new security protocols.
- Explanation A2 (predicted probability - 0.72%): One of the most contributing features describing a different behavior is the $non\_working\_days\_http$, which indicates that the entity has activity outside working hours. This could represent that users use their corporate laptop or smartphone for personal uses or keeps the device always connected.
- Explanation A3 (predicted probability - 0.64%): The most notorious features in this explanation are the usage of applications or services using a deprecated protocol (SSL version 1.1). This behavior might indicate that the user is using deprecated software or applications, which might be affected by vulnerabilities.
- Explanation A4 (predicted probability - 0.83%): The final proposed explanation includes a high contribution of the $http\_feature\_request\_body\_len\_ratio$ feature, which might indicate that the entity usually downloads a high amount of information from web servers.

After the assessment of local explanations, it is possible to align specific countermeasures and long-term strategies to reduce the exposition risk in the case that a given user is exposed to the behaviors identified in A1-A4, which are root causes of phishing cases. Presented below is a repertoire of mitigation measures for explanations A1 to A4.

- Remediation R1 (matching explanation A1): Review DNS configuration to avoid vulnerabilities and check for software updates.
- Remediation R2 (matching explanation A3): Identify the software and the reasons why a deprecated protocol is being used. Check for updates and force applications to use TLS v1.2 or v1.3.
- Remediation R3 (matching explanation A2 and A4): Create an awareness security policy to remind company's assets (e.g., laptops) should be used to access and download only trusted resources and use them only when necessary (i.e., during working hours).

1) Considerations: In this project, the usage of SHAP is proposed due to its stability and model-agnostic explanations. Also, SHAP is particularly well-suited for explaining predictions of tree-based algorithms like XGBoost (the algorithm used in the Risk Assessment Framework). Furthermore, this popular library allows us to compute local explanations.

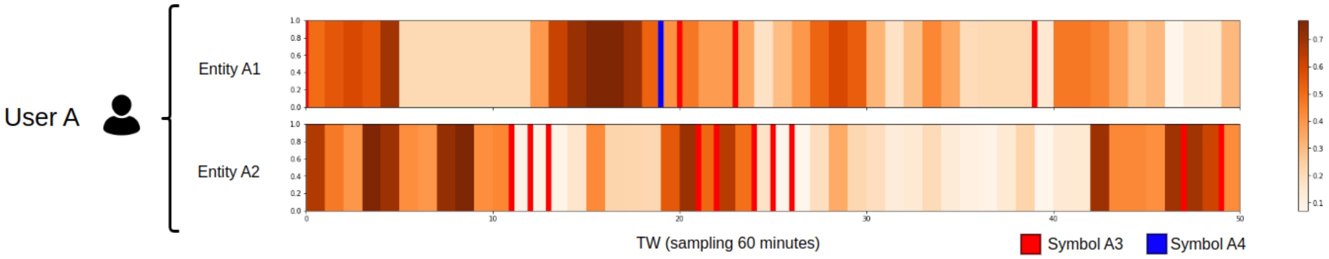


Fig. 5. For each user entity, the predicted probability is calculated for each TW in the production data, and the alignment is computed using evaluation metrics. The TW corresponds to a time period of two consecutive days (Feature Agreement, k:5)

TABLE II
SUBSET OF FEATURES THAT ARE THE MOST CONTRIBUTING FEATURES IN EXPLANATIONS A1-A4 AND B1

Feature	Description
$http\_common\_ua\_ratio$	Mean of popular and expected User Agent values in HTTP events - not included in the initial feature list
$http\_response\_body\_len\_ratio$	Mean of response body length event in the sequence. A high value could indicate the entity is downloading data.
$http\_request\_body\_len\_ratio$	Mean of request body length event in the sequence. A high value could indicate data exfiltration.
$http\_status\_200\_ratio$	Ratio of HTTP events with status code 2xx indicating success
$mean\_interlog\_dns\_time$	Mean value of the time between two consecutive logs in the TW
$dns\_qtype\_used\_ratio$	Ratio of query types used in DNS events
$dns\_qtype\_obsolete\_ratio$	Ratio of query types which are label as obsolete
$dns\_common\_udp\_ports\_ratio$	Ratio of common UDP ports used. Unusual port numbers could indicate malicious activity
$dns\_recursion\_desired\_ratio$	Ratio of DNS events with recursion flag (RD) set
$mean\_interlog\_ssl\_time$	Mean value of the time between two consecutive SSL logs in the TW
$ssl\_version\_ratio\_v10$	Ratio of events using SSL version v1.1
$ssl\_interlog\_time\_0.001$	Time between two logs with less than 0.001 seconds
$non\_working\_days\_http$	Boolean set to true when HTTP activity is generated by an entity in non working days
$non\_working\_days\_ssl$	Boolean set to true when SSL activity is generated by an entity in non working days

B. Explanation Evaluation

The systematic evaluation of explanations is conducted through the utilization of faithfulness metrics. These metrics were introduced in the works of [34] [35]. Our proposal is based on utilizing faithfulness metrics, enabling a systematic comparison of the production explanations with the ground truth explanations knowledge base. Specifically, the objective of using these metrics is to provide a quick mapping between the ground truth information, thereby reducing the analysis efforts in production.

In detail, the evaluated disagreement metrics are defined as follows: the Feature Agreement metric (9), which is based on determining the shared group of features between explanation E in the production data and explanation G in the ground truth data within the top k positions. The Rank Agreement metric (10) resolves disagreements by determining matching features between the two explanations that share the same symbols. Lastly, the Signed Rank Agreement metric combines both feature and rank agreement to determine disagreements between explanations based on both rank and sign.

$$FeatureAgreement(E,G,k) = \frac{|top(E,k) \cup top(G,k)|}{k} \tag{9}$$

$$\frac{\cup_{s \in S} |s \in |top(E,k) \cup top(G,k) \wedge rank(E,k) = rank(G,k)|}{k} \tag{10}$$

$$\frac{\cup_{s \in S} |s \in |top(E,k) \cup top(G,k) \wedge sign(E,k) = sign(G,k)|}{k} \tag{11}$$

The process of evaluating explanations is illustrated in Figure 5. For each entity, post-hoc explanations are computed using the SHAP procedure as explained previously. Each timestamp (TW) included in the validation dataset is assessed using the symbols stored in the Ground Truth Database, with a fixed value of k for each assessment. In the diagram, aligned symbols are represented using blue and red vertical lines corresponding to aligned symbols A3 and A4, which correspond to the usage of applications or services employing deprecated protocols and a high volume of downloads from web servers, respectively. From the proposed visualization, it is evident that users frequently employ a deprecated TLS protocol, primarily from the workstation (Entity A2). Additionally, traces from smartphone devices (Entity A1) are also observed. Furthermore, we propose a general overview in the form of a ranking system that provides stakeholders with an organic interface, offering suggestions for mitigating risks

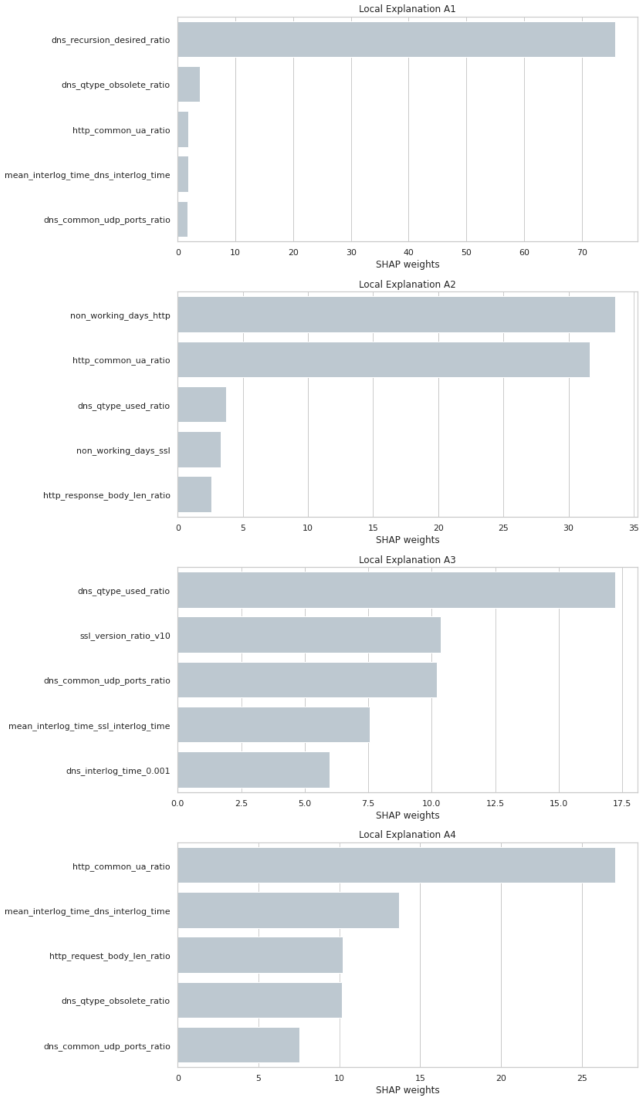
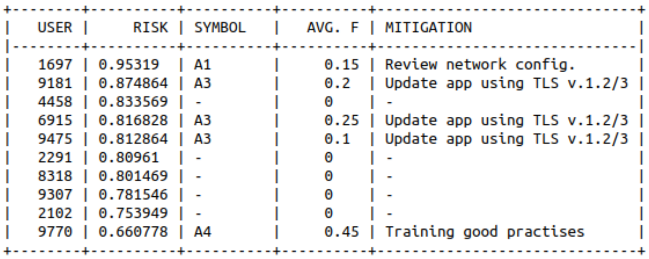


Fig. 6. Local Explanations of different TW.



```
+--------+----------+----------+----------+------------------------------+
|   USER |     RISK | SYMBOL   |   AVG. F | MITIGATION                   |
|--------+----------+----------+----------+------------------------------|
|   1697 | 0.95319  | A1       |     0.15 | Review network config.       |
|   9181 | 0.874864 | A3       |     0.2  | Update app using TLS v.1.2/3 |
|   4458 | 0.833569 | -        |     0    | -                            |
|   6915 | 0.816828 | A3       |     0.25 | Update app using TLS v.1.2/3 |
|   9475 | 0.812864 | A3       |     0.1  | Update app using TLS v.1.2/3 |
|   2291 | 0.80961  | -        |     0    | -                            |
|   8318 | 0.801469 | -        |     0    | -                            |
|   9307 | 0.781546 | -        |     0    | -                            |
|   2102 | 0.753949 | -        |     0    | -                            |
|   9770 | 0.660778 | A4       |     0.45 | Training good practises      |
+--------+----------+----------+----------+------------------------------+
```

Fig. 7. An illustrative example of the produced risk score. For each prediction, the alignment is computed using the Ground Truth Database.

among users who show strong statistical evidence (risk) of being compromised in a threat campaign. Figure 7 includes an example of the top k users for a specific threat campaign. For each user positioned at the top of the ranking, we include the following information: the risk (mean predicted accuracy of all the user's entities), the symbol that corresponds to the most frequent symbol occurrence, the 'average f' value describing the frequency, and finally, the proposed mitigation. In this article, the proposed mitigations are simply the mapping of symbols to mitigation strategies, as detailed in Section IV-A.

1) Considerations: Improving the Groundtruth Explanations Knowledge System: After analyzing the diagram presented in Figure 7, it becomes evident that there are regions with a high predicted probability for the two entities that lack an aligned symbol. This phenomenon arises due to the limited number of symbols in the Groundtruth Explanations database. This is due, for the scope of this article, we have constructed a database consisting of only four distinct symbols, each associated with three different mitigation approaches. The labeling approach involves selecting random explanations from the set TW with a high predicted probability. To iterative enhance the knowledge system, we propose that stakeholders select explanations with high accuracy and manually assess them using the procedure described in Section IV-A. Figure 8 illustrates the second explanation for $Entity A2$, which has a predicted probability of 0.76 in identifying behavior associated with phishing.

Upon a thorough analysis of the provided explanation, it is discerned that the most influential factors contributing to this prediction are the utilization of common UDP communications for data transmission and the signal code "200," signifying a successful status response from an HTTP protocol communication. Under this scrutiny, it is concluded that the analyzed explanation does not contain any instances of non-legitimate behavior. Nevertheless, the presence of these symbols could be attributed to model bias, thereby offering opportunities for improving classification models.

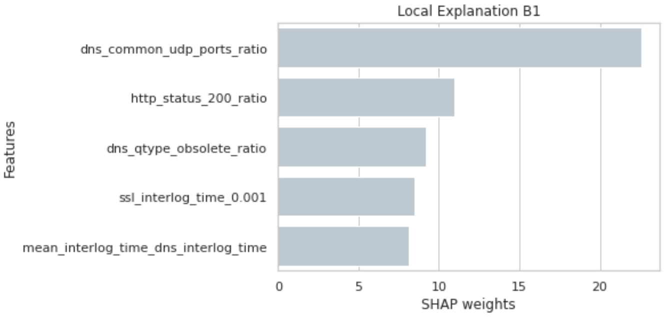


Fig. 8. Local Explanation of a TW

2) Considerations: Limitations on the Metric Selection: In this study, we empirically evaluate the three proposed metrics. The signed agreement was the metric which best matches our scenario using a fixed k of 5. To elaborate further, the first metric, known as Feature Agreement, focuses solely on determining if two features align in the highest score. However, within the cybersecurity domain, whether a feature contributes positively or negatively to constructing an explanation can carry distinct implications. To this end, the Sign Agreement metric maintains this aspect, offering a more organic comparison between two explanations. Lastly, the signed-rank agreement metric was excluded from consideration as stakeholders did not observe a clear distinction in the weightings of top positions within an explanation.

V. Conclusion and Discussion

In this study, we propose the utilization of disagreement metrics for the systematic assessment of the fidelity of explanations, as demonstrated through a successful use case in the cybersecurity domain. As expected, the usage of faithfulness metrics could be beneficial to reduce fatigue in assessing explanations, as the the proposed methodology enables the alignment with known behaviors. Furthermore, we believe that this mechanism allows security professionals to share the labeled behaviors amongst SOC teams in a threat-sharing fashion. For instance, the labeled exposition behaviors could be shared, allowing enterprises to provide prevention capabilities in cold start configurations. Moreover, the proposed approach has the potential to significantly enhance threat behavior profiling by systematically evaluating explanations. This can facilitate Tactics, Techniques, and Procedures (TTPs) correlation from the MITRE ATT&CK framework and enable the construction of kill-chains to provide a comprehensive understanding of attacker strategies. This represents a promising step towards threat prediction and hunting.

Regarding the proposed explainability proxy, the future work of the framework hinges on determining efficient heuristics for generating ground-truth explanations. On the other hand, while the proposed metrics facilitate comparisons between ground-truth explanations and production explanations, these comparisons are rigid and primarily based on the characteristic of relevance in top-ranking positions. We believe that these metrics could be extended to incorporate uncertainty and establish a dynamic scoring system based on the relevance of the explanation.

Acknowledgments

This work has been supported by Smart Catalonia actions in collaboration with the Cybersecurity Agency of Catalonia, as well as the valuable support of the entire i2CAT team.

References

[1] U. Comission, "Proposal for a REGULATION OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL LAYING DOWN HARMONISED RULES ON ARTIFICIAL INTELLIGENCE (ARTIFICIAL INTELLIGENCE ACT) AND AMENDING CERTAIN UNION LEGISLATIVE ACTS," 2021. [Online]. Available: https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:52021PC0206

[2] D. Gunning and D. Aha, "DARPA's Explainable Artificial Intelligence (XAI) Program," *AI Magazine*, vol. 40, no. 2, pp. 44–58, Jun. 2019. [Online]. Available: https://ojs.aaai.org/aimagazine/index.php/aimagazine/article/view/2850

[3] A. Calvo, S. Escuder, J. Escrig, M. Arias, N. Ortiz, and J. Guijarro, "A Data-driven Approach for Risk Exposure Analysis in Enterprise Security," in *2023 IEEE 10th International Conference on Data Science and Advanced Analytics (DSAA)*, Oct. 2023, pp. 1–9. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/10302480

[4] F. Doshi-Velez and B. Kim, "Towards A Rigorous Science of Interpretable Machine Learning," Mar. 2017, arXiv:1702.08608 [cs, stat]. [Online]. Available: http://arxiv.org/abs/1702.08608

[5] L. H. Gilpin, D. Bau, B. Z. Yuan, A. Bajwa, M. Specter, and L. Kagal, "Explaining Explanations: An Overview of Interpretability of Machine Learning," Feb. 2019, arXiv:1806.00069 [cs, stat]. [Online]. Available: http://arxiv.org/abs/1806.00069

[6] C. Molnar, *Interpretable Machine Learning*. leanpub.com, 2022. [Online]. Available: https://christophm.github.io/interpretable-ml-book/

[7] B. Coma-Puig, A. Calvo, J. Carmona, and R. Gavaldà, "A case study of improving a non-technical losses detection system through explainability," *Data Mining and Knowledge Discovery*, Apr. 2023. [Online]. Available: https://doi.org/10.1007/s10618-023-00927-7

[8] J. N. Paredes, J. C. L. Teze, G. I. Simari, and M. V. Martinez, "On the Importance of Domain-specific Explanations in AI-based Cybersecurity Systems (Technical Report)," Aug. 2021, arXiv:2108.02006 [cs]. [Online]. Available: http://arxiv.org/abs/2108.02006

[9] W. Min, W. Liang, H. Yin, Z. Wang, M. Li, and A. Lal, "Explainable Deep Behavioral Sequence Clustering for Transaction Fraud Detection," Jan. 2021. [Online]. Available: http://arxiv.org/abs/2101.04285

[10] Y. Lin, R. Liu, D. M. Divakaran, J. Y. Ng, Q. Z. Chan, Y. Lu, Y. Si, F. Zhang, and J. S. Dong, "Phishpedia: A Hybrid Deep Learning Based Approach to Visually Identify Phishing Webpages," in *30th USENIX Security Symposium (USENIX Security 21)*. USENIX Association, 2021, pp. 3793–3810. [Online]. Available: https://www.usenix.org/conference/usenixsecurity21/presentation/lin

[11] P. Barnard, N. Marchetti, and L. A. DaSilva, "Robust Network Intrusion Detection Through Explainable Artificial Intelligence (XAI)," *IEEE Networking Letters*, vol. 4, no. 3, pp. 167–171, Sep. 2022.

[12] I. Psychoula, A. Gutmann, P. Mainali, S. H. Lee, P. Dunphy, and F. Petitcolas, "Explainable Machine Learning for Fraud Detection," *Computer*, vol. 54, no. 10, pp. 49–59, Oct. 2021.

[13] S. Wali and I. Khan, "Explainable AI and Random Forest Based Reliable Intrusion Detection system," Dec. 2021.

[14] A. Adadi and M. Berrada, "Peeking Inside the Black-Box: A Survey on Explainable Artificial Intelligence (XAI)," *IEEE Access*, vol. 6, pp. 52 138–52 160, 2018. [Online]. Available: https://ieeexplore.ieee.org/document/8466590/

[15] F. Charmet, H. C. Tanuwidjaja, S. Ayoubi, P.-F. Gimenez, Y. Han, H. Jmila, G. Blanc, T. Takahashi, and Z. Zhang, "Explainable artificial intelligence for cybersecurity: a literature survey," *Annals of Telecommunications*, vol. 77, no. 11, pp. 789–812, Dec. 2022. [Online]. Available: https://doi.org/10.1007/s12243-022-00926-7

[16] D. K. Sharma, J. Mishra, A. Singh, R. Govil, G. Srivastava, and J. C.-W. Lin, "Explainable Artificial Intelligence for Cybersecurity," *Computers and Electrical Engineering*, vol. 103, p. 108356, Oct. 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0045790622005730

[17] M. Wang, K. Zheng, Y. Yang, and X. Wang, "An Explainable Machine Learning Framework for Intrusion Detection Systems," *IEEE Access*, vol. 8, pp. 73 127–73 141, 2020, conference Name: IEEE Access.

[18] T. Zebin, S. Rezvy, and Y. Luo, "An Explainable AI-Based Intrusion Detection System for DNS Over HTTPS (DoH) Attacks," *IEEE Transactions on Information Forensics and Security*, vol. 17, pp. 2339–2349, 2022, conference Name: IEEE Transactions on Information Forensics and Security.

[19] M. Sarhan, S. Layeghy, and M. Portmann, "Evaluating Standard Feature Sets Towards Increased Generalisability and Explainability of ML-Based Network Intrusion Detection," *Big Data Research*, vol. 30, p. 100359, Nov. 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2214579622000533

[20] R. Alenezi and S. A. Ludwig, "Explainability of Cybersecurity Threats Data Using SHAP," in *2021 IEEE Symposium Series on Computational Intelligence (SSCI)*, Dec. 2021, pp. 01–10.

[21] R. R. Karn, P. Kudva, H. Huang, S. Suneja, and I. M. Elfadel, "Cryptomining Detection in Container Clouds Using System Calls and Explainable Machine Learning," *IEEE Transactions on Parallel and Distributed Systems*, vol. 32, no. 3, pp. 674–691, Mar. 2021, conference Name: IEEE Transactions on Parallel and Distributed Systems.

[22] P. Giudici and E. Raffinetti, "Explainable AI methods in cyber risk management," *Quality and Reliability Engineering International*, vol. 38, no. 3, pp. 1318–1326, 2022. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/qre.2939

[23] M. Zolanvari, Z. Yang, K. Khan, R. Jain, and N. Meskin, "TRUST XAI: Model-Agnostic Explanations for AI With a Case Study on IIoT Security," *IEEE Internet of Things Journal*, vol. 10, no. 4, pp. 2967–2978, Feb. 2023, conference Name: IEEE Internet of Things Journal.

[24] W. Guo, D. Mu, J. Xu, P. Su, G. Wang, and X. Xing, "LEMNA: Explaining Deep Learning based Security Applications," in *Proceedings of the 2018 ACM SIGSAC Conference on Computer and Communications Security*, ser. CCS '18. New York, NY, USA: Association for Computing Machinery, Oct. 2018, pp. 364–379. [Online]. Available: https://dl.acm.org/doi/10.1145/3243734.3243792

[25] H. Liu, C. Zhong, A. Alnusair, and S. R. Islam, "FAIXID: A Framework for Enhancing AI Explainability of Intrusion Detection Results Using Data Cleaning Techniques," *Journal of Network and Systems Management*, vol. 29, no. 4, p. 40, May 2021. [Online]. Available: https://doi.org/10.1007/s10922-021-09606-8

[26] M. N. Al-Mhiqani, R. Ahmad, Z. Z. Abidin, K. H. Abdulkareem, M. A. Mohammed, D. Gupta, and K. Shankar, "A new intelligent multilayer framework for insider threat detection," *Computers & Electrical Engineering*, vol. 97, p. 107597, Jan. 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0045790621005322

[27] J. Lee, J. Kim, I. Kim, and K. Han, "Cyber Threat Detection Based on Artificial Neural Networks Using Event Profiles," *IEEE Access*, vol. 7, pp. 165 607–165 626, 2019, conference Name: IEEE Access.

[28] H. Mei, G. Lin, D. Fang, and J. Zhang, "Detecting vulnerabilities in IoT software: New hybrid model and comprehensive data analysis," *Journal of Information Security and Applications*, vol. 74, p. 103467, May 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2214212623000510

[29] Y. Izza, A. Ignatiev, and J. Marques-Silva, "On Explaining Decision Trees," Oct. 2020, arXiv:2010.11034 [cs]. [Online]. Available: http://arxiv.org/abs/2010.11034

[30] V. Kecman, "Support Vector Machines – An Introduction," in *Support Vector Machines: Theory and Applications*, ser. Studies in Fuzziness and Soft Computing, L. Wang, Ed. Berlin, Heidelberg: Springer, 2005, pp. 1–47. [Online]. Available: https://doi.org/10.1007/10984697

[31] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, Oct. 2001. [Online]. Available: https://doi.org/10.1023/A:1010933404324

[32] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '16. New York, NY, USA: Association for Computing Machinery, Aug. 2016, pp. 785–794. [Online]. Available: https://dl.acm.org/doi/10.1145/2939672.2939785

[33] S. M. Lundberg and S.-I. Lee, "A Unified Approach to Interpreting Model Predictions," in *Advances in Neural Information Processing Systems*, vol. 30. Curran Associates, Inc., 2017.

[34] 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," Feb. 2022, arXiv:2202.01602 [cs]. [Online]. Available: http://arxiv.org/abs/2202.01602

[35] C. Agarwal, S. Krishna, E. Saxena, M. Pawelczyk, N. Johnson, I. Puri, M. Zitnik, and H. Lakkaraju, "OpenXAI: Towards a Transparent Evaluation of Model Explanations," Jan. 2023. [Online]. Available: http://arxiv.org/abs/2206.11104