Journal of Network and Systems Management (2021) 29:40
https://doi.org/10.1007/s10922-021-09606-8

FAIXID: A Framework for Enhancing AI Explainability of Intrusion Detection Results Using Data Cleaning Techniques

Hong Liu[1] · Chen Zhong[2] · Awny Alnusair[1] · Sheikh Rabiul Islam[3]

Received: 16 October 2020 / Revised: 20 March 2021 / Accepted: 22 April 2021 /
Published online: 24 May 2021

Abstract

Organizations depend on heavy use of various cyber defense technologies, including intrusion detection and prevention systems, to monitor and protect networks and devices from malicious activities. However, large volumes of false alerts from such technologies challenge cybersecurity analysts in isolating credible alerts from false positives for further investigations. In this article, we propose a framework named FAIXID that leverages Explainable Artificial Intelligence (XAI) and data cleaning methods for improving the explainability and understandability of intrusion detection alerts, which in turn assist cyber analysts in making more informed decisions fueled by the quick elimination of false positives. We identified five functional modules in FAIXID: (1) the pre-modeling explainability module that improves the quality of network traffic's data through data cleaning; (2) the modeling module that provides explanations of the AI models to help analysts make sense of the model internals; (3) the post-modeling explainability module that provides additional explanations to enhance the understandability of the results produced by the AI models; (4) the attribution module that selects the appropriate explanations for the analysts according to their needs; and (5) the evaluation module that evaluates the explanations and collects feedback from analysts. FAIXID has been implemented and evaluated using experiments with real-world datasets. Evaluation of results demonstrates that the utilization of data cleaning and AI explainability techniques provide quality explanations to analysts depending on their expertise and backgrounds.

Keywords AI explainability · Intrusion detection · Data cleaning · AI-based cybersecurity · Cyber threat intelligence

Extended author information available on the last page of the article

1 Introduction

With the continuous increase of network throughput and cybersecurity threats, intrusion detection not only challenges cyber defense technologies but also challenges computing power and human investment. The network traffic generated from an organizational network is overwhelming. Although Intrusion Detection Systems (IDS) have been a popular solution, one of the prevailing problems of IDS is the very high rate of false positives that overwhelm cybersecurity analysts with false IDS alerts. This is a labor-intensive and expensive process [1, 2]. One of the main directions of IDS research relies on adopting Artificial Intelligence (AI) technologies to reduce the workload of analysts. However, AI technologies can be limited by the training data, and existing models may not perform well in dynamic cyberspace. Besides, the lack of explainability of AI results usually hinders human analysts from trusting them. The fact that AI technologies and human analysts cannot collaborate effectively has resulted in three urgent issues in intrusion detection system management: (1) the rapid growth of network traffic data volume to consider for analysis, (2) the lack of explanatory data analysis results, and (3) labor-intensive and expensive analysis process caused by false alarms.

In addressing these challenges, it is important to realize that cybersecurity analysts of different roles have various expectations on the results of AI technologies associated with the management of intrusion detection systems. Therefore, a holistic view of the human-in-the-loop analysis process is needed for managing the AI technologies and results in IDSs.

To provide a holistic view, we propose a framework that incorporates data cleaning and Explainable Artificial Intelligence (XAI) technologies into the data analysis process of intrusion detection to facilitate the collaboration between human analysts and technologies. Data cleaning has been widely used to improve the quality of data. XAI technologies have been used in IDS to improve the explainability and understandability of intrusion detection alerts [3, 4], which in turn assist cyber analysts in making more informed decisions by quickly eliminating false positive alerts.

One of the major challenges for leveraging AI and data analysis algorithms is to create a data set that contains relevant and accurate information. The complex nature of real-world data makes data quality problems more evident because errors may spread and lead to useless or even harmful results. This explains why data cleaning has received increasing attention in recent years [5–8]. The cleaning process is critical because it bridges the gap between the data we want and the data we have, and it directly affects the processes that come after it. In this article, several typical data cleaning techniques have been employed to obtain relevant high-quality data for intrusion detection systems, including quality assessment, aggregation, sampling, and feature selection techniques.

Furthermore, we need XAI to improve fairness, accountability, and trust in decisions. According to Lipton et al. [9], XAI has three different notions: (1) pre-modeling explainability—using a cleaned and/or summarized set of features, (2) in-modeling explainability—generating explanation and prediction together, and (3)

post-modeling explainability—explaining the dynamics between inputs and outputs in an already trained/tested AI model.

To this end, we developed a novel framework named FAIXID that has been designed and implemented for the purpose of enhancing the explainability of AI-powered intrusion detection alerts. FAIXID provides a system structure that identifies the functional modules critical for providing cybersecurity analysts with more accurate and explainable results in the setting of intrusion detection. These functional modules include the *Pre-Modeling Explainability Module*, *Modeling Module*, *Post-Modeling Explainability Module*, *Attribution Module*, and *Explanation Evaluation Module*.

This work uniquely contributes to the proper linking of the established field of intrusion detection/prediction and the re-emerging fields of data cleaning and AI explainability by developing an integrated framework that assists analysts with different backgrounds and levels of expertise to make better decisions regarding threat identification and mitigation. More specifically, the primary contribution of this work is twofold:

- FAIXID is an integrated framework that incorporates data cleaning and AI explainability techniques in the data analysis process of intrusion detection. FAIXID employs data cleaning methods to filter out the noise data in the dataset through data quality assessment, data measurement, and filtration of low-quality data. Following data cleaning and construction of a subset of a high-quality and relevant dataset, FAIXID employs Explainable AI algorithms to ensure that the resulting AI models are better interpretable and understandable to cybersecurity analysts.
- FAIXID emphasizes the importance of understanding the different needs of human analysts in the human-in-the-loop data analysis process. In addition to the explainability and modeling modules, we defined the attribution module and evaluation module in FAIXID as two important functional modules: the *Attribution Module* selects the appropriate algorithms for the analysts according to their needs, and the *Evaluation Module* evaluates the explanations provided to analysts based on task performance, user experience, and explainability functions.

This article is organized as follows. In Sect. 2, we provide background information and an overview of previous research on cybersecurity analysts, intrusion detection and prediction, and explainable artificial intelligence. In Sect. 3, we present detailed information about the structure and components of our framework. Sect. 4 describes the evaluation procedure, setup and the roles of FAIXID modules in the evaluation process. In Sects. 5 and 6, we describe the evaluation experiments using human subjects and proxy methods, respectively. Finally, the paper concludes in Sect. 7 with some future research directions.

2 Background Knowledge and Literature Survey

There has been quite a significant body of research in intrusion detection, prediction, and response techniques; many of these research efforts have been implemented into useful tools. However, there are few research works that address the utilization of Explainable Artificial Intelligence (XAI) and data cleaning techniques for improving the explainability of intrusion detection results. Hence, they are more explainable and understandable by human analysts. In this section, we categorize recent research efforts in these areas and discuss them separately.

2.1 Cybersecurity Analysts in SOC

SOCs (Security Operations Centers) are responsible for monitoring and analyzing real-time activities within the organizational network. It employs various technology solutions, such as IDS/IPS (Intrusion Detection/Prevention Systems), firewalls, log systems, SIEM (Security Information and Event Management) systems, etc. While working with technology solutions, cybersecurity analysts investigate the alerts and logs in real-time to detect malicious cyber incidents. D'Amico and Whiteley investigated the roles of Computer Network Defense Analysts. They outlined the analysis stages, including triage analysis, escalation analysis, correlation analysis, threat analysis, incident response, and forensic analysis [1]. The hierarchical analysis process is also reflected in the Cocoa, a SOC ontology for analysis, which is aligned to the NIST cybersecurity framework [10]. The lower-level analysis (usually done by Tier-1 analysts) involves monitoring and investigating the incoming alerts and reports generated by IDS/IPS and SIEM systems for possible malicious activities [2]. The lower-level analysis reports are escalated to upper-level analysis to further understand patterns of attack and make predictions about future attacks. The analysts' experience and domain knowledge play an important role in these stages of analysis [11, 12].

The real-time alerts and logs are overwhelming for analysts to process because of many false positives (i.e., false alerts and reports). Therefore, AI approaches have been brought to the field to reduce analysts' workload and increase their work efficiency [13, 14]. Given the fact that analysts are coupled with AI-enabled solutions, the decisions from AI solutions have to be explainable to gain analysts' trust and help analysts in making a confident and accountable decision. Analysts of different responsibilities have different requirements from AI explainability. For example, Tier-1 analysts may care more about the logic of a model in filtering and correlation analysis and whether the logic is reasonable for confident decision making.

2.2 Cybersecurity Intrusion Detection and Prediction

Intrusion detection is mainly about detecting and mitigating cybersecurity threats. We begin this section by reviewing recent intrusion detection approaches, focusing on techniques that utilize Data Mining (DM) and Machine Learning (ML). Later in

this section, we provide a review of recent approaches that are more focused on intrusion prediction. While both intrusion detection and prediction mechanisms are helpful, FAIXID differs from these efforts because it does not assume that the data is high-quality. Instead, it employs data cleaning techniques to smooth out the noise in the dataset that it operates on. Then it utilizes AI Explainability methods to make the results of IDS alerts more explainable and understandable to human analysts.

ML and its sibling DM techniques focus on training and testing the data and revealing its hidden properties. Recently, Peng et al. [15] proposed a clustering-based intrusion detection technique based on an improved version of the K-means algorithm in order to handle big datasets. The proposed method is combined with the use of Principal Component Analysis (PCA) to reduce the dimensionality of the processed dataset and improve clustering efficiency. Experimental evaluation of the approach was demonstrated on the KDDCUP99 dataset, which shows that clustering based on mini-batch K-means along with PCA can be efficient for intrusion detection over big datasets. For voluminous data in Wireless Sensor Networks, Otoum et al. [16] proposed an ML-based approach for intrusion detection that leverages Reinforcement Learning (RL) methods. The approach shows that RL-based intrusion detection outperforms other approaches in terms of detection performance, accuracy, and precision.

Uwagbole et al. [17] proposed an ML approach for SQL Injection Attack (SQLIA) detection and prevention in the context of big data. The authors utilized a predictive analytics web application with vast quantities of learning data to train a classifier. The goal is to detect SQLIA in web requests before they mature and reach the back-end database. The primary objective that this approach achieved in comparison with other related works is the scalability factor, where the approach worked well and achieved good performance and scores with massive quantities of data.

In Smart Cities, connected vehicles continuously communicate and exchange information, which makes them prone to cyber-attacks. As such, researchers have recently focused on intrusion detection of connected vehicles in smart city environments. Most recently, Aloqaily et al. [18] proposed an automated cloud-based framework for intrusion detection that maintains user's requirements in terms of quality of service and quality of experience. As such, the proposed hybrid method (called D2H-IDS) is meant to distinguish between trusted service requests vs. false requests during an intrusion attack. In this approach, decision tree ML is used for data classification purposes, which helped the system to achieve high accuracy and detection rates while reducing the number of false negatives and false positives.

Goeschel [19] proposes a model for reducing false positives in intrusion detection systems based on DM methods by combining support vector machines (SVM), decision trees, and Naïve Bayes algorithms. The data was initially divided into normal/abnormal using SVM. A model based on decision trees in this approach is used to identify known attacks, while a Naïve Bayes classifier is used to identify unknown attacks. Using these three algorithm types, the false alarm rates were significantly reduced. Hachmi et al. [20] presents an optimization method named MOP-IDS that uses multiple Intrusion Detection Systems to identify false positives and false negatives based on a three-step process. The first step is filtering the low-

level alerts. In the second and third steps, clustering techniques are used to reduce redundancy, and then binary optimization is used to detect false positives and false negatives in the produced set of alerts.

Gil Perez et al. [21] presents the design of a collaborative Intrusion Detection Network System (IDS) model, which uses a reputation model to assess IDS alerts for the purpose of detecting false IDS alerts and distributed attacks in multiple environments. The role of the reputation model is to ensure that the alerts are being assessed for trustworthiness and credibility. In terms of security risk management and mitigation, most recently, Khosravi-Farmad1 et al. [22] , presents an integrated framework that models the necessary information required for network security risk management. This framework is based on a probabilistic graphical model named Bayesian Decision Network (BDN). Additionally, a cost-benefit concrete analysis is performed to promote risk mitigation. This is accomplished by employing a Variable Elimination (VE) inference algorithm to account for budget limitations.

In terms of mitigating efforts, Otoum et al. [23] proposed a technique for mitigating false negatives using a two-tier intrusion detection approach in Wireless Sensor Networks as being used in Smart Grid applications. The proposed system utilizes two subsystems–first, an anomaly detection subsystem with Enhanced Density-Based Spatial Clustering of Applications with Noise method. Second, incident signature detection is being used with Random Forest method. The utilization of these two subsystems shows that the number of false negatives is reduced when the rate is higher of the anomaly detection subsystem, and it is lower of the signature detection subsystem.

While intrusion detection is concerned with discovering and mitigating threats as described previously, cybersecurity intrusion prediction is focused on taking proactive measures to understand network vulnerabilities to proactively harden and improve the system's resilience against potential cybersecurity incidents. AI techniques, including Machine Learning, Deep Learning, Data Mining (DM), and Natural Language Processing (NLP), have been extensively utilized to predict cybersecurity incidents allowing analysts to respond to such incidents before the actual damage happens. As such, new trends have recently emerged, focusing on analyzing datasets to obtain knowledge that can be used to train classifiers to forecast cybersecurity incidents. Liu et al. [24] proposed a technique that is based on analyzing the properties of an organization's network data flow to forecast cybersecurity incidents. To do so, they have utilized a large number of features that are obtained from network mismanagement symptoms and malicious activity time series such as spam and phishing. These features are then used to train an Random Forest (RF) classifier against incident reports. The relative importance of features has been demonstrated along with the classifier performance and accuracy in forecasting cybersecurity incidents.

Soska and Christin [25] proposed an ML classifier that can predict whether a website may become compromised and pose security risks in the near future before it actually becomes malicious. The proposed classification system utilizes a large corpus of websites for training purposes, and a set of static features were obtained automatically from the Alexa Web Information Service and web page content. It is clear that the performance of ML-based methods relies heavily on the selection of

features that the classifier needs to base its decisions on. This is a significant drawback, especially when domain knowledge is not appropriately utilized to identify a highly appropriate set of features. Furthermore, relying only on website content and traffic statistics to predict if the site will be malicious at some point is quite risky and can be violated easily[25].

Security threats and incidents are usually saved in natural language. Therefore, NLP-based tools [26–28] have been proposed in the literature to assist in detecting and predicting cybersecurity incidents. Such tools can provide good prediction performance when the utilization of domain knowledge is done in a meaningful manner to process data. For example, Ritter et al. [26] proposed an approach that focuses on discovering focused security events on Twitter by using a weakly supervised seed-based extraction technique. Using a small number of seed examples, automatic extractors for new categories of security events are defined and trained in a weakly supervised manner. The system has been demonstrated in three categories of security events, namely, DoS attacks, account hijacking, and data breaches. Generally speaking, techniques that employ NLP to process textual data in social media and elsewhere can be considered as being highly domain-specific, and it will be quite hard to adapt such approaches to other domains with a slightly different vocabulary.

Other ML/DM approaches focus on predicting malicious websites [25, 29]. Borgolte et al. [29] proposed a system named Delta System based on static analysis that can be used to identify malicious activities and infection campaigns in a website based on the observed modifications and differences between the previous and the current version of the site. The point is to facilitate the identification and removal of infections and mitigate additional future infections by determining if the new version of the website is benign or malicious. The analysis in the proposed Delta system begins by retrieving and normalizing both versions of the website. Based on this normalization, the similarities between the two versions of the site are computed using a fuzzy tree difference algorithm that performs the tree-to-tree comparison. The process continues by clustering the assignment of the similarity vector. Finally, the identifying signature of known infection campaigns is generated.

Wang et al. [30] proposed an explainable ML-based framework for IDS. The approach uses Shapley Additive explanations as well as local and global explanations in order to improve the explainability of intrusion detection systems. This is something similar to our approach. However, there are many fundamental differences. Specifically, in this approach, the role of human analysts is the user who reviews the IDS results but not a component in the intrusion detection process. In our work, we consider human analysts as an important player in the process. As we will discuss later in this paper, human analysts of different responsibilities have different needs for the explanations. Therefore, a holistic view of the human-in-the-loop analysis process is needed for managing the AI technologies and results in IDSs. To provide this holistic view, our framework incorporates data cleaning and Explainable Artificial Intelligence (XAI) technologies into the data analysis process of intrusion detection to facilitate the collaboration between human analysts and technologies. One important contribution of our framework is that it identifies the attribution module, the evaluation modules, and two important functional modules

that take human analysts' needs and feedback as the processing inputs. The Attribution Module selects the appropriate algorithms for the analysts according to their needs, and the Valuation Module evaluates the explanations provided to analysts based on task performance, user experience, and explainability functions. These two modules are one of the primary differences between our works and many related existing works.

2.3 Explainable Artificial Intelligence

Explainable AI (XAI) is a re-emerging field of study, after the earlier work of [31, 32], and [33], that focuses on understanding an AI model by interpreting and then explaining its contents so it can be easier for human end-users to understand it and reason about it. Previous work primarily focused on explaining the decision process of knowledge-based systems and expert systems. Recent advancements in AI and ML, their application to diverse areas, and concerns over unethical use and undesired biases in the models are some of the top reasons behind the renewed interests in XAI research. In addition, recent concerns and laws by different governments are necessitating more research in XAI. In addition, in February 2020, the U.S. Department of Defense (DoD)[1] adopted ethical principles for Artificial Intelligence [34] that encompasses five major areas: responsible (exercising an appropriate level of judgment in AI capabilities), equitable (minimizing unintended bias in AI capabilities), traceable (transparent and auditable AI capabilities), reliable (explicit and well-defined uses of AI capabilities), and governable (ability to detect and avoid unintended consequences, and ability to deactivate deployed systems).

Yang et al. [35] adopt the concept of *Bayesian Teaching* that uses a subset of examples selected by the domain experts instead of using the entire dataset. However, selecting the right subset of examples from the real-world is somehow challenging. Lei et al. [36] propose an approach for sentiment analysis that uses a subset of text as an explanation of prediction instead of the entire text. However, their approach is only limited to text-based analysis. [37] propose a model agnostic explanation technique that explains the prediction of a black box model using a simple and interpretable model in the local context. They emphasize putting humans in the loop for enhancing trust in the decision.

Kim et al. [38] propose an interpretability approach that helps quantify the sensitivity of prediction to high dimensional concepts, such as the concept of "striped" can be utilized for identifying the image of Zebra from images not containing Zebra. The concept is usually constructed from a user-defined set of examples. Furthermore, for better explainability and validation of results from neural network based black-box models, Horel et al. [39] develop a statistical test to assess the statistical significance of the features/variables in a single layer feed-forward neural network. In addition, the test statistics also enable one to rank the variables based on their influence on the prediction. However, their approach is limited to only a single layer feed-forward neural network and requires a very large amount of samples to fit the data in the asymptotic distribution.

[1] https://www.defense.gov

Mariano et al. [40] applied an adversarial approach to finding minimum modification of the input features of an intrusion detection system needed to reverse the classification of the misclassified instance. Besides satisfactory explanations of the reason for misclassification, their approach is model agnostic and can be extended to further diagnosis. Furthermore, [41] work on understanding the implications of adversarial samples on Recurrent Neural Network (RNNs) as an IDS since RNNs are good for sequential data analysis and network traffic exhibits some sequential patterns. They find that adversarial the adversarial training procedure can significantly reduce the attack surface.

Furthermore, [42], propose an explainable Deep Neural Network framework for anomaly detection in industry settings. Their approach explains questions like why something is an anomaly and what is the confidence of the explanation. In the industrial control system, an alarm from the intrusion/anomaly detection system has a very limited role unless the alarm can be explained with more information. [43] design a layer-wise relevance propagation method for DNN to map the abnormalities between the calculation process and features. This process helps to compare the normal samples with abnormal samples for better understanding with detailed information. More recently, decentralized and distributed "Plug and Play" (PnP) AI tools are becoming more attractive because of the vast number of Internet of Things (IoT) devices, and an enormous amount of data. Ridhawi et al. [44] envision a novel general AI solution that automatically selects appropriate dataset, model (e.g., supervised, unsupervised), and configurations (e.g., neural network configuration), and recognizes the data set (i.e., understand data type, provides data reasoning).

Most of these available approaches find the deviation from the base/average scenario. Lime [37] tries to generate an explanation from local behavior by approximating the model with an interpretable model (e.g., decision trees, linear model). However, Lime is limited by the use of only a linear model to approximate the local behavior. Furthermore, [45] propose "SHAP" that combines functionalities from previous seven approaches: LIME [37], DeepLIFT [46], Tree Interpreter [47], QII [48], Shapley sampling values [49], Shapley regression values [50], and Layer-wise relevance propagation [51] to explain the prediction in a model agnostic way. While SHAP comes with solid theoretical background from game theory, usually it is computationally intensive [52]. ELI5 also uses the LIME algorithm internally for explanations. However, ELI5 is mostly limited to tree-based and other parametric or linear models. Similarly, Tree Interpreter is limited to only tree-based approaches such as Random Forest and Decision Trees.

Different users of AI models may have various purposes of explainability in models in terms of trustworthiness, causality, transferability, informativeness, confidence, fairness, accessibility, interactivity, and privacy awareness[53]. Therefore, addressing these issues needs more attention.

3 Framework for Enhancing Explainability of Intrusion Detection

Since IDS alerts are read and interpreted by human cybersecurity analysts, there is a dire need to reduce or eliminate the amounts of false positives that are usually produced by IDS alert systems and consequently enhance the explainability of intrusion detection results. To this end, we have developed a novel framework named FAIXID that integrates AI explainability methodologies and data cleaning techniques in the domain of IDS/IPS alerts analysis. The proposed framework is general enough to accommodate different scenarios and applications by not subscribing to a particular specification or technological solution. As illustrated in Fig. 1, FAIXID includes five functional modules: *Pre-modeling Explainability Module*, *Modeling Explainability Module*, *Post-modeling Explainability Module* and an *Attribution Module* and *Explanation Evaluation Module*. These integrated modules work together to provide cybersecurity analysts of different roles with more accurate and more explainable data that help them with their daily decision-making practices.

In the following sub-sections, we discuss the modules in the proposed framework with their functional components.

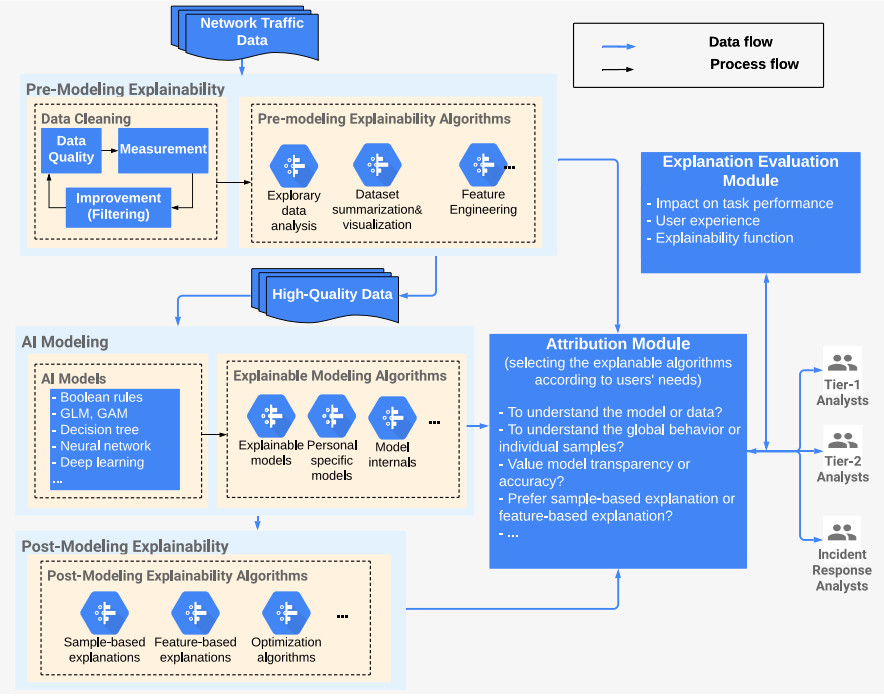


Fig. 1 FAIXID: the framework for enhancing explainability of intrusion detection

3.1 Pre-modeling Explainability Module

The purpose of this module is to improve data quality, understand and describe the data used to develop the model. This module contains two primary functional components: data cleaning and pre-modeling explainability algorithms.

3.1.1 Data Cleaning

The data cleaning component is designed to improve the quality of the incoming network traffic data by processing the data in order to detect irrelevant pieces and cleanse inaccurate data through a three-phase approach, which includes *data quality*, *measurement*, and *improvement* phases. Considering the large volume of network data and the high rate of false alerts of IDS, data cleaning is essential for intrusion detection.

High-quality data can help cybersecurity analysts quickly and accurately determine suspicious network activities to detect potential attacks. In addition, the quality assessment allows analysts to have a comprehensive understanding of data resources, which will help to use them systematically, and integrate them into the detection process.

The *data quality* phase attempts to understand the nature of the data source and relevant context information in order to identify high-quality data from the incoming raw traffic. This is accomplished by utilizing relevant data quality assessment strategies. Based on our previous works in the domain of data quality assessment for data cleaning [54], we utilize four of the most relevant dimensions for identifying high-quality data from a raw dataset: accuracy, completeness, consistency, and timeliness. In the *measurement* phase of data cleaning, a scoring system is used to measure the quality of data based on pre-defined quality metrics. In general, multiple metrics can be associated with each quality dimension. In the *improvement* phase of the data cleaning component, we utilize sampling methods in order to filter out low-quality data and obtain a subset of high-quality data that can be fed to the next phase for processing. In most situations, we only need a portion of the most relevant high-quality data. Therefore, the goal of filtering is to convert large raw data into a small subset of the most pertinent data that is most useful for a particular application.

We summarize the processing phases of the data cleaning component as follows:

1. Phase 1 - Data Quality Definition: The input of this phase is a data source and context information. In the experimentation section of this paper (Sect. 4), we use the UGR network traffic data. Data Analysis aims to identify relevant data quality dimensions in the given context. The output of this phase is a list of the quality dimensions.
2. Phase 2 - Measurement: The input at this phase is the dimensions of data quality obtained from the first phase. This phase performs quality measurement according to the definition of quality metrics. The output is the data quality value and quality-related issues.

3. Phase 3 - Improvement: Uses the data quality value obtained in the second phase as input, and compares it with the quality requirement value to identify the critical areas, and then different methods are used in this stage to improve the quality of data.

3.1.2 Pre-modeling Explainability Algorithms

The pre-modeling explainability component is a collection of explainability algorithms that intend to achieve a common goal of gaining a better understanding of the dataset used for the development of AI models. Examples of algorithms utilized in this component include exploratory data analysis, feature engineering, dataset summarization and visualization methods, and other relevant algorithms.

Exploratory Data Analysis (EDA) is used to explore the data, and extract a summary of the main features and characteristics of a given dataset. Dataset summarization aims to find a minimal subset of representative samples that can represent the essence of the dataset. Interpretable feature engineering includes domain-specific and model-based feature engineering. The goal of feature engineering algorithms is to extract a useful set of features that help us understand and interpret the data based on domain knowledge. Such feature extraction is a crucial step as future steps in the framework and ML algorithms heavily rely on how carefully these features are selected, and how well the domain knowledge is being applied during the process of extracting representative features. The implementation details of feature selection are described in Sect. 4. Collectively, the algorithms in the pre-modeling explainability component help cybersecurity analysts in making sense of the data features by producing higher quality data.

3.2 Modeling Explainability Module

This module contains AI models and the explanations that enable users of this framework to make sense of the AI models and enable user-driven explanations for decision-making purposes. The explainable in-modeling algorithms such as Boolean Rule Column Generation(BRCG) [55] algorithm described in Sect. 4.3 are mainly concerned with model interpretability that provides transparent models and AI model internals (e.g., parameters, weights, feature interaction, etc.) to allow users to understand the model. Infusing domain knowledge in the model for better explainability has also been tried successfully by [3, 56] . Models can have different levels of transparency, which have been carefully described in [53]. As such, this module is carefully designed to enhanced modeling explainability by empowering users of the system to understand the reasoning behind the results generated by the models and thus making more sound decisions.

3.3 Post-modeling Explainability Module

The post-modeling explainability module provides a collection of various algorithms that are designed to provide additional information about how an AI

model produces a particular result. For example, sample-based explanations methods examine the dataset and identify the most influential representative training samples/examples to explain machine learning results. We used the Contrastive Explanations Method (CEM) [57] to provide sample-based explanations, which is described in Sect. 4.3. Feature-based explanation algorithms explain the results of black-box models, and enhance model interpretability by ranking the features in the dataset according to an assigned importance value. Besides, optimization algorithms can be used to generate explanations about how deep learning models work internally [58].

3.4 Attribution Module

Given the variety of explainability algorithms provided in the pre-modeling, modeling, and post-modeling modules, the *Attribution Module* selects the appropriate algorithms for the analysts according to their needs for different levels of explainability. In addition, it examines the attributes of the data from different angles and selects the right explanatory functions to provide analysts with varying levels of explainability that suit their needs. The selection of the explainability level that satisfies a given user is accomplished based on answering questions such as the ones shown in Fig. 1. In Sect. 5, we have conducted experiments that demonstrate the novelty of this attribution module when selecting the right explanatory algorithm for analysts based on their needs and roles.

3.5 Explanation Evaluation Module

The explanations provided to the analysts are evaluated by the *Explanation Evaluation Module*. Human evaluation is the most widely used evaluation method. Drawing on the taxonomy of Machine Learning interpretability proposed by Doshi-Velez and Kim [59], we propose to evaluate the explanations from three perspectives: impact on task performance, user experience, and explainability function. The impact on task performance and user experience can be evaluated by human subjects, and the explainability function can be evaluated based on quantitative metrics, such as faithfulness[60] and monotonicity[61].

4 Framework Evaluation

The proposed explainability framework, FAIXID, has been implemented into an integrated prototype. We used this prototype to test and evaluate the framework's modules on intrusion detection scenarios using network traffic analysis.

The primary objectives of the evaluation include:

- To evaluate whether the modules in our framework can fulfill the needs of cybersecurity analysts of various roles
- To evaluate whether data cleaning increases the explainability of IDS results
- To examine what explanations serve analysts the best, and to investigate the analysts' needs through FAIXID's evaluation module

Table 1 Different types of network flows in the selected UGR'16 dataset [62]

Type	Description
background	Background traffic
blacklist	Traffic involving the IP addresses included in the public blacklists
botnet	Botnet traffic including bots behaviors such as sending SPAM, connecting to C&C server, and performing click fraud.
dos	Denial of service traffic that includes SYN packets sent to the victims
scan11	One-to-one port scanning
scan44	Four-to-four port scanning
spam	SMTP spam traffic
sshscan	SSH scanning attack
udpscan	UDP scanning attack

– To investigate whether a non-human subject-based explainability quantification method (i.e., proxy-based method) can be applied to quantify the quality of explanation.

In this section, we discuss the setup and processes that we utilized to evaluate the framework considering the nature of the dataset, the data cleaning and filtering process, feature selection process, and explainability algorithms, as described in FAIXID modules. In the following section, we discuss the human-subject experiments that we conducted using this setup. In Sect. 6, we discuss how we evaluated the explainability of the framework using proxy methods.

4.1 Experiment Setup

We adopt a network traffic dataset, named UGR'16 [62], in our intrusion detection evaluation tasks. This dataset is a collection of real network traffic for about four months, from a tier-3 Internet Service Provider (ISP), containing background and attack traffics. It is a well-labeled dataset with the necessary ground truth of attack. We selected a portion of the dataset (i.e., 115 GB), which includes the network flows captured in the time window of August 1, 2016, to August 7, 2016. The selected dataset contains the most types of network flows within a week, as shown in Table 1.

In this malware version of the dataset, infected bots send SPAM, connect to an HTTP C&C server, and use HTTP to perform some *ClickFraud*[2].

The features of the dataset includes the timestamp of the end of a network flow ("Timestamp"), duration of network flow ("FlowDuration"), source IP address ("SrcIP"), destination IP address ("DstIP"), source port ("SrcPort"), destination port ("DstPort"), protocol ("Protocol"), forwarding status ("ForwardingStatus"),

[2] A type of fraud occurs online in pay-per-click advertising.

type of service ("TypeofService"), packets exchanged ("ExedPacketss"), and the corresponding bytes ("Bytes").

We apply AI algorithms to detect the attack-related traffic in the data set, which mimics the AI-based intrusion detection systems. According to the proposed framework, we provide explanations in the pre-modeling, modeling, and post-modeling phases to help analysts understand the analysis result better. We apply an open-source software toolkit, AIX360 [63] to generate explanations in different contexts. Furthermore, we evaluate the explanations in three use cases targeting different analysts. The explanations are evaluated from the aspects of impact on task performance, user experience, and explainability functions. The results are presented in Sect. 5.

4.2 Data Cleaning

For data cleaning, the most commonly used data quality dimensions include completeness, accuracy, timeliness, etc. Through the pre-analysis of the UGR dataset, including completeness and repeatability assessment, the analysis results show that UGR data has a high degree of completeness and low repeatability. Therefore, considering the results and the context of the data set, when defining data quality, we pay more attention to two dimensions—timeliness, and relevancy. — Timeliness is an important aspect of data, especially for the intrusion detection scenario. If certain types of network traffics are from a certain period, these network activities have a higher probability of being a potential attack. Furthermore, — Relevancy refers to the degree of relevance between network activities and known data, such as blacklist data. If a network activity is associated with blacklist data, then this traffic has a higher probability of being a potential attack and therefore has a higher data quality.

4.2.1 Filtering Low-Quality Data

In most situations, we only need a portion of the most relevant high-quality data. Therefore, the goal of filtering is to convert a large raw dataset into a small subset of the most relevant data that is most useful for a particular application. In the filtering stage, we use sampling methods to obtain high-quality data.

We aggregate the data set by combining the hourly data. First, we group the data by source IP, source port, destination IP, destination port, and protocol. Second, we calculate the number of occurrences, forwarding status, type of service, and the average of exchanged packets and bytes. The aggregated dataset was reduced to 57.12 GB from 115.11 GB (original dataset).

Furthermore, due to the large scale of the dataset, we use a two-step sampling method to extract high-quality sample data. First, we apply the random sampling method to weekly data, which provides an unbiased representation of the dataset since each member of the dataset has an equal probability of being chosen. At the same time, we get another dataset with 418,593 records based on related datasets (i.e., blacklist) using a two-step sampling method. A blacklist dataset contains

abnormal network flow data. Finally, we merge two datasets to obtain the final dataset with 1,083,606 records applied in AIX360.

4.2.2 Feature Selection

We remove the "ForwardingStatus" feature because it has a constant value in the dataset. Besides, we remove the feature "Bytes" following a correlation analysis (Pearson Correlation) among features as correlated features overfit the model, and one of the correlated features is enough for the analysis. The correlation matrix in Fig. 2 demonstrates that the feature "Bytes" is highly correlated with the feature "PacketExed", considering the threshold = 0.8 used for identifying strong correlations.

After data cleaning, we trained a Boolean Rule Column Generation (BRCG) model to predict normal/abnormal network traffic, and the accuracy of predicting abnormal network traffic based on the cleaned data is 0.87.

4.3 Explainability Algorithms

The AIX360 toolkit supports a list of explainability algorithms, including ProtoDash, Disentangled Inferred Prior VAE, Contrastive Explanations Method, Contrastive Explanations Method with Monotonic Attribute Functions, LIME, SHAP, Teaching AI to Explain its Decisions, Boolean Decision Rules via Column Generation, Generalized Linear Rule Models, and ProfWeight. In this work, we use the following four algorithms to generate explanations that support the three cases described in Sect. 5.2. We use the Boolean Rule Column Generation(BRCG) [55] algorithm which provides a direct interpretable supervised learning method for

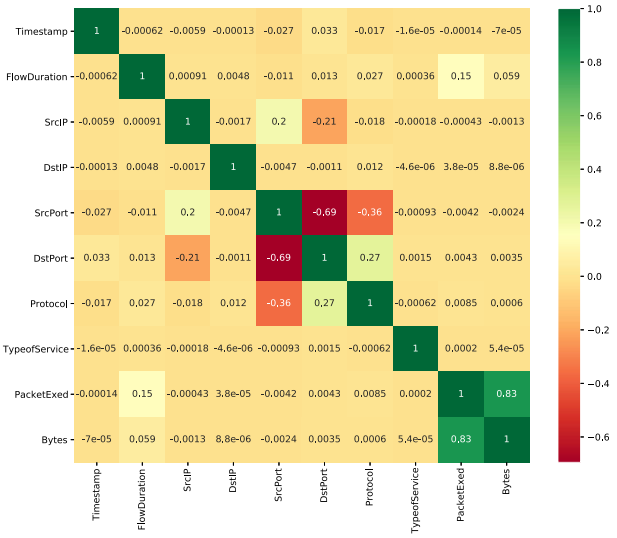


Fig. 2 Correlation matrix of the features

binary classification. This algorithm uses the column generation technique to learn a Boolean rule in disjunctive normal form or conjunctive normal form. The Logistic Rule Regression(LogRR) [64] is a supervised learning method that can be directly interpreted, which can perform logistic regression on rule-based functions.

ProtoDash [65] algorithm provides example-based explanations for summarizing datasets and explaining predictions from an AI model. It uses a fast gradient-based algorithm to find prototypes, examples that best summarizes and compactly represents the data distributions, along with their importance weights. We use the Contrastive Explanations Method (CEM) [57] to compute contrastive explanations which highlights both pertinent positive (PP) and pertinent negatives (NP). This algorithm finds minimally sufficient (pertinent positive) and the necessarily absent (pertinent negatives) to maintain the original classification.

5 Evaluation of Explainability using Human Subjects

In this section, we provide details about the most recent significant experiments that we have designed and conducted to evaluate the proposed framework. In particular, we have conducted a human-subject experiment that intends to evaluate the effectiveness of the explainability provided to cybersecurity analysts in terms of task performance, user experience, and the explainability function. These aspects have been identified in the *Explanation Evaluation Module* of the framework.

The human-subject experiment was carried out based on the setup that was discussed in the previous section. The experiment was conducted in controlled settings with the consideration of task complexity. The goal is to evaluate how well potential cybersecurity analysts perform real-world analysis tasks using the explainability enhancements of IDS alerts provided by our framework.

The primary hypotheses that we set out to test in this experiment include:

H 1 The proposed framework provides cybersecurity analysts with quality explanations that assist them in making better decisions when trying to understand and identify potential incidents.

H 2 FAIXID provides explanations that are helpful, sufficient, and easy to understand, for analysts, depending on their roles and skills.

5.1 Participants

We screened the participants based on their domain knowledge in machine learning, computer networking, and intrusion detection. Seven human subjects participated in our study. Table 2 shows participant's levels of knowledge of ML techniques, fundamental computer networking, and intrusion detection.

5.2 Procedure and Case Descriptions

The impacts of explanations on task performance can be evaluated by comparing the analysts' task performance with and without explanations. Due to the small

Table 2 The knowledge level of the human subjects

Subject	Machine Learning	Computer Networking	Intrusion Detection
S1	◖	●	◖
S2	◖	●	●
S3	●	●	●
S4	●	◖	◖
S5	●	●	●
S6	●	◖	◖
S7	●	●	●

○: low, ◖: medium, ●: high

number of subjects involved in this study, we adopted the forward simulation approach[59] in which subjects were presented with the explanations and model input (i.e., a piece of network flow data), and they were asked to simulate the output (i.e., whether it is attack traffic or simply a false positive). This procedure allows us to understand if the explanations help the subjects to understand the model output.

Besides, we presented three cases to the participants and asked them to take a different role of analysts in each case to evaluate how the framework can meet different needs of analysts: the first case targets data triage analysts who want to improve their understanding of data models; the second case targets the analysts who are interested in evaluating the prediction results; and the third case targets the analysts with the responsibility of incident response who are interested in the importance of data features on the prediction results.

User experiences were measured by the rating of the perceived helpfulness, explanation sufficiency, and the ease of understanding: helpfulness refers to how helpful a participant finds an explanation is in terms of his/her task; explanation sufficiency asks how well the provided explanation meets the participant's need; and the ease of understanding asks how easy it is to understand the explanation. On a 5-point Likert Scale questionnaire, subjects express their agreement, neutrality, and disagreement through 5-4, 3, and 2-1 points, respectively (Fig. 3).

Case 1: Model Explainability:

As explained in Sect. 2.1, the analysts who take care of data triage would prefer to understand the model as a whole. They need to compare the models based on their domain knowledge and experience and present their findings to upper-level

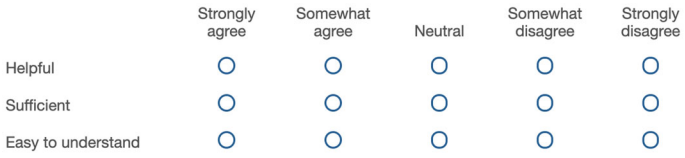


Fig. 3 A rating question about the perceived helpfulness, sufficiency, and ease of understanding

analysts. A good understanding of the model enables analysts to recognize overfitting problems and bias in the model.

In this case, we chose the interpretable models for classification, including the BRCG and the LogRR model. Because of the simple forms, the analysts can quickly understand the logic of the model and learn how the classification results were produced. We also have observed a trade-off between model simplicity and prediction accuracy. In our case study, participants were provided with the visualization of how the prediction results are linearly related to the numerical features and rules.

Figure 4 shows some examples of visualized explanations provided to analysts to understand the dependence of the LogRR model on individual features. These explanations visualize the generalized additive model components, including first-degree rules and linear functions of the features.

Case 2: Prototypical Analysis:

Analyzing prototypes helps analysts understand the AI model predictions by reviewing the representatives that are similar to a network flow being predicated as abnormal/normal. The analysts of various levels of analysis (e.g., triage, escalation, and correlation analysis) may be interested in reviewing and understanding the prototypical network flows that have been classified as a normal or abnormal activity. We used the ProtoDash algorithm [65] implemented in AIX360 to obtain and present prototypes to our participants.

The participants were asked to play the role of analysts whose responsibilities include reviewing the reports from data triage analysts and correlating incidences to understand the potential threats. In this case, the selected prototypical explanations were provided to the participants to help them understand the prediction results. One explanation includes one selected network flow which was identified as normal traffic and the top five prototypes with similar features to the selected one. The other explanation includes the selected abnormal network flow and its top five prototypes with similar features. The top prototypes were selected using the Protodash algorithm [65]. Table 3 shows one example of the explanations.

Case 3: Feature-Based Analysis:

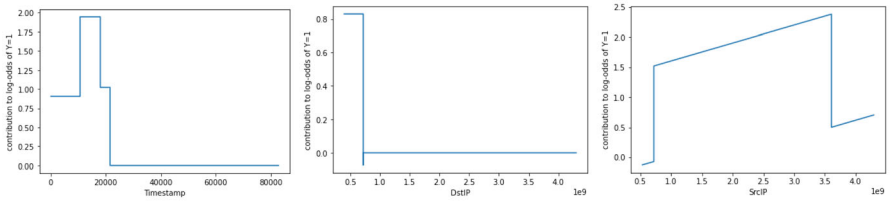


Fig. 4 Example explanations that visualize how the AI model (i.e., the logistic rule regression model) depends on the individual features. The plots include the generalized additive model components, excluding the higher-degree rules. The Y-axes refer to the contribution to the contribution to the log-odds of the prediction of abnormal network flow. The values of the features, including Timestamp, DstIP, and SrcIP have been converted to the ordinal numbers and been binarized, as required by the logistic rule regression model

Table 3 One sample explanation in Case 2, showing the top prototypes with the similar features of a selected abnormal network flow. The values in the cell are the importance weight in the range of (0, 1), which indicates how similar the feature of the prototype is to the selected one

Feature	Prototype1	Prototype2	Prototype3	Prototype4	Prototype5
Timestamp	1.00	0.56	0.15	0.50	0.44
FlowDuration	1.00	1.00	1.00	0.08	1.00
SrcIP	1.00	1.00	0.72	0.32	0.07
DstIP	0.96	0.08	0.08	0.19	0.08
SrcPort	1.00	0.20	0.39	0.39	0.39
DstPort	1.00	1.00	0.08	0.89	0.39
Protocol	1.00	1.00	1.00	0.08	1.00
TypeofService	1.00	1.00	1.00	1.00	1.00
PacketExed	1.00	1.00	1.00	0.08	1.00

Analysts with the responsibility of incident response and threat analysis may not care about how the IDS/IPS models work precisely. Instead, they would prefer a more abstract black-box view of the model to focus on learning how features impact the decisions.

In this case, the participants were asked to play the role of analysts of incident response and pay special interest in why a network flow was identified as abnormal, and if so, what changes made in feature values would let them be identified as normal. Therefore, a constructive explanation can provide the information needed for this type of analyst.

In our experiment, we used the constructive explanation algorithm in AIX360, which is developed by [57]. Two explanations are demonstrated in Fig. 5. The left explanation shows the feature importance in terms of pertinent negatives (PNs), and the right one shows the feature importance of pertinent positives (PPs). To calculate PNs, the algorithm first identifies the network flows that have been predicated as abnormal and then changes the values of a minimal set of features that would yield a different prediction result (normal). On the other hand, PPs are calculated by

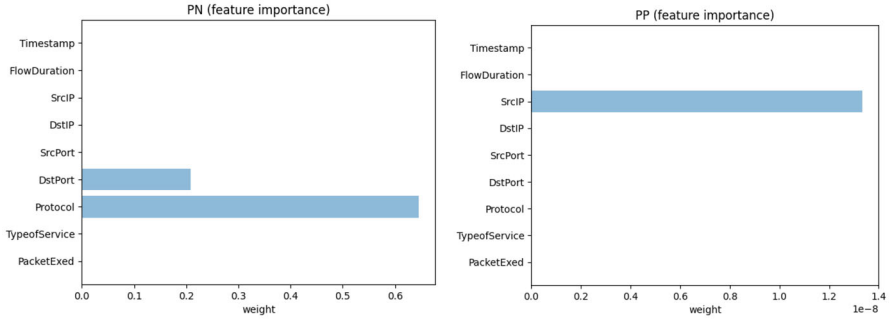


Fig. 5 Example explanations of feature importance in Case 3

identifying the minimal set of features that would change the prediction results from normal to abnormal.

The PNs and PPs explain how the prediction results could have been different through minimal changes to features and help analysts understand the importance of features. For instance, given the left plot in Fig. 5, an analyst can tell the Protocol and DstPort are the key features for the IDS model and changes to the values of these features may yield different prediction results. The analyst will check whether it is consistent with their expectation based on their expertise. Otherwise, it may indicate a disagreement between the judgment of analysts and the IDS model, and the model needs to be updated.

5.3 Discussion of the Rating Responses

Recall that the user experience was measured by the rating of the perceived helpfulness, explanation sufficiency, and ease of understanding in Fig. 3, the boxplots of the rating scores in the three cases are shown in Fig. 6. These experiments allowed us to gain several inspiring findings. First of all, most participants agreed that the explanations provided in the cases were helpful. We noticed that there were three participants in Case 2 rated low ("Disagree") for "Helpfulness' and the "Ease of Understanding". It makes sense that the explanations can be found less useful when participants were unable to completely understand them.

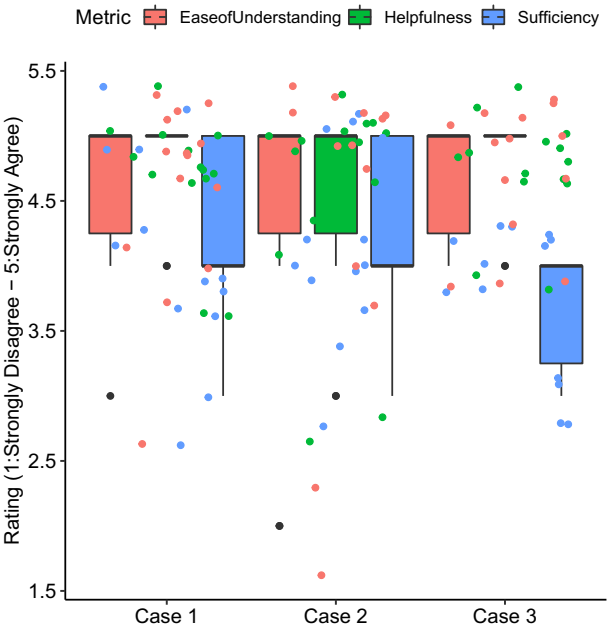


Fig. 6 User study: rating responses of the three cases - the dots in red, green, and blue are the rating response; the black dots are the outliers of the boxplots

Although the number of participants and the number of cases used in this study are not very large to provide conclusive data with statistical evidence about FAIXID's effectiveness, the data shown in Fig. 6 is promising, and it suggests that FAIXID can meet different needs of analysts when they try to identify potential incidents via explanations that are overall helpful, sufficient, and easy to understand. As a part of our future work, we plan to conduct a comparison experiment to collect more evidence to show the FAIXID's internal mechanisms can help analysts in becoming more productive.

Another observation is that most explanations were found useful by the participants, but the rating of "Sufficiency" is relatively low, especially in Case 3. These results indicate that participants expected more explanations when taking the analysts' role in Case 3. This observation is consistent with our assumption that analysts of various roles (described in Sect. 5) have different needs for explanations. To satisfy the different needs, the attribution module in FAIXID plays a critical role, and it works as a recommendation system to suggest which explanations may serve the analysts the best in a given intrusion detection task. As such, the value of using the attribution module provides partial support for hypothesis H2.

In this experiment, the attribution module was implemented manually such that we decided which explanations can meet the needs of analysts in each case prior to the case study and select XAI algorithms to generate these explanations. Therefore, these explanations provided to subjects were generated according to their roles in each case. We believe that a refined attribution module could improve the rating of "Sufficiency". The ratings of "Sufficiency" suggest that the attribution module needs to consider the feedback of the analysts as an important input to make more relevant recommendations to analysts. Developing an attributions system that can autonomously collect analysts' feedback as well as the requirements of an intrusion detection task is currently being investigated as a possible future enhancement for FAIXID. We expect that such an attribution system would make more concrete recommendations of the relevant and useful explanations to analysts of different roles.

5.4 Threats to Validity

Our human-subject experiments exercised various aspects of our framework. Some of these aspects come with their limitations and threats to validity. Therefore, there exist several external threats to the validity of our exploratory experiments. Firstly, the validity of the experiments is limited by the choice of the dataset used in the study. While the selected subset of the UGR dataset that we used in our study consists of real traffic data, and is carefully chosen to include background traffic and attack traffic, the network flow is well-labeled, and the time span of this traffic is four months. Secondly, although we have informally experimented with some other datasets and obtained comparable results, the experiments we reported in this article are from one dataset. Performing more comprehensive empirical evaluations with other datasets can strengthen our conclusions and reduce some of these threats. Additionally, while the size of the dataset used in our experiment is considered

acceptable, we cannot establish the scalability and performance bench-marking of the used algorithms without testing it on bigger datasets.

Finally, although the three use cases with human-subjects gave us good insights into the usefulness of our framework in terms of how helpful, sufficient, and easy to understand the explanations were, the use cases are quite limited by the number of subjects. As such, a larger-scale user study with more subjects in which some of them are professional cyber analysts may provide us a better insight into the quality of the explanations provided. Ideally, we would like to have the same subjects experiment with tasks that use the proposed framework and other tasks without it. However, it is quite hard to design tasks that have the exact level of complexity that allows us to draw useful conclusions about the effectiveness of the proposed framework. In such a user experience evaluation, it would undoubtedly mitigate many of the threats described previously.

Since user experiments involve human-subjects, such experiments usually introduce internal threats to validity because the subjects have varying degrees of background and proficiency in cybersecurity analysis. More user experiments that involve a considerably larger number of participants would certainly mitigate these threats.

6 Evaluation of Explainability using Proxy Methods

In this section, we provide detailed information about the quantitative explainability experiment designed and conducted to evaluate the proposed framework. We conducted this experiment based on the proxy methods to quantify the effectiveness of data cleaning in terms of explainability. This aspect has been identified in the *Explanation Evaluation Module* of the framework. The hypothesis we set out to test in this experiment is outlined below:

H 3 The utilization of data cleaning techniques improves the quality of the incoming data traffic in manners that enhances the explainability of intrusion detection.

Quantification of explainability is still an open challenge and far from the expectation. In our prior work [66], we propose a proxy task-based explainability quantification method for XAI in *credit default prediction*. In this work, we apply the same approach [66], but for a different case, for the quantification of explainability in XAI for the *intrusion detection*. A proxy task-based explainability quantification method considers different properties of output representation, such as the depth of a decision tree, and size of rule list as a metric for evaluation of explainability. Humans have a limit on the capacity of processing information—average human can process 7 ± 2 pieces of information (i.e., cognitive chunks) to understand something [67]. According to [66], in the most generalized form, the quality of an explanation depends upon the number of cognitive chunks or information pieces that the recipient of explanation has to relate to in order to understand an explanation (i.e., the less, the better) of a prediction. A summary of their approach is as follows.

Lets assume, E = explainability; N_c = number of cognitive chunks; I = interaction; N_i = number of input cognitive chunks; and N_o = number cognitive chunks or information pieces involved in the explanation representation, i.e., output cognitive chunks.

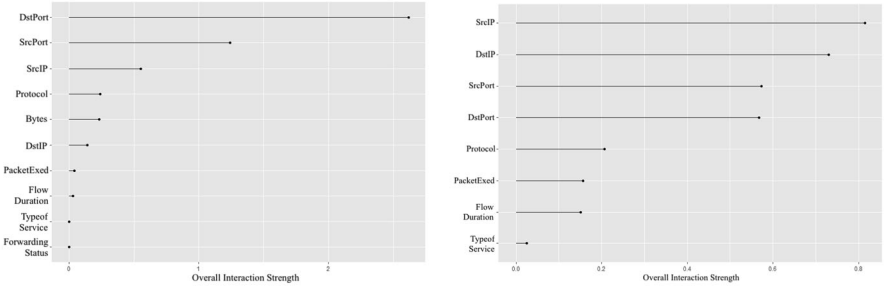
$$E = \frac{1}{N_c} \tag{1}$$

However, the interaction among cognitive chunks, from the correlated features, complicates the explainability. Therefore, the Formula 1 is penalized for having an interaction among cognitive chunks, resulting in Formula 2.

$$E = \frac{1}{N_c} + (1 - I) \tag{2}$$

where, the interaction I ranges in between 0 and 1. We use the Programming language R's *iml* package, which uses the *partial dependence* of individual features as the basis, to measure the interaction (i.e., interaction strength) (I) among features. Usually, the less the interaction, the better the explainability, so the Formula took the complement of I. Figure 7a represents the feature interaction before data cleaning, and Fig. 7b represents the feature interaction after data cleaning. We use the *iml* package from the programming language *R* to generate the interaction strength among features. From Fig. 7a and 7b, it is evident that our data cleaning reduces the interaction among features which ultimately will result in better explainability of output.

Furthermore, both input cognitive chunks and output cognitive chunks are important to understand the causal relationship between input and output. It is also vital for good explanations. The ideal case to avoid the correlation problem is to have one input and one output cognitive chunk, but this is rare and unusual in real-world cases. After the segregation of input and output cognitive chunks, Formula 2 can be re-written as Formula 3:



(a) Before data cleaning.

(b) After data cleaning.

Fig. 7 Interaction strength before and after data cleaning

$$E = \frac{1}{N_i} + \frac{1}{N_o} + (1 - I) \tag{3}$$

where N_i refers to the number of input cognitive chunks, and N_o refers to the number of cognitive chunks involved in the explanation representation. Usually, explanations become more complicated with the increase of cognitive chunks. For both input and output, the ratios of the best case (i.e., one cognitive chunk) and observed case (for a particular application), $1/N_i$ and $1/N_o$, are added towards the total explainability measure.

To commensurate the importance of different predicates, a weight term (i.e., w) is added for each of three predicates. Therefore, Formula 3 becomes Formula 4:

$$E = \frac{w_1}{N_i} + \frac{w_2}{N_o} + w_3(1 - I) \tag{4}$$

Formula 4 can then be used to quantify the explainability of the explanation method (i.e., global explainability) for any classifier. We can use Formula 4 to also quantify the local explainability too—explainability of an instance-level prediction. In that case, only the first predicate of Formula 4 (including the weight term) remains the same (i.e., the same number of input chunks). However, the remaining two predicates, predicate 2 and predicate 3, will vary from instance to instance as an interaction strength (I) depends on the involved cognitive chunks in the representation of explanation for a particular instance.

Applying Formula 4, on metadata (Table 4) of two different feature settings (before and after data cleaning), we see that *clean data* provides the best explainability score of 0.2056, which is a multitude of improvements compared to the 0.066 that we get using the *raw data* (Table 4). In fact, performance for any post-hoc interpretability/explainability method will still be limited to 0.0085 if it does not reduce the number of cognitive chunks to represent the output. The results of the table show that the data cleaning mechanism is indeed increasing the explainability score from 0.066 to 0.2056, which will ultimately help to obtain better explainability output results. Therefore, it provides strong support for our hypothesis H3.

Despite some signs of progress in explainability quantification methods, there are still some challenges that need more attention. Some of the challenges include: (1) constructing an optimal approach for calculating instance-level feature contribution that takes correlations among features into considerations due to the matter that the correlation among feature affects as well as complicates explanations, and (2) reaching into an agreement of what an explanation is and to whom, and a formalism

Table 4 Explainability score from raw data and clean data

	Raw Data	Clean Data
Input chunks (N_i)	10	8
Output chunks (N_o)	8	8
Int. Strength (I)	1.00	0.6327
Explainability (E)	0.0666	0.2056

for the explanation [68] highlighted the diverse and dynamic explainability requirements of users in AI/ML ecosystems with a case scenario to advance the research of accessible explainable AI.

7 Conclusion

Cyber attacks have been growing in volume and complexity near exponentially. This leaves cybersecurity analysts overwhelmed with the process of navigating through the countless number of daily IDS threat alerts, most of which are not considered very credible. The proposed FAIXID framework uniquely combines data cleaning techniques and XAI in a single framework that intends to assist analysts in their daily threat monitoring activities.

We evaluated the framework using different case studies, including human-analyst experiments and a proxy evaluation experiment. The result of human-subject experiments, demonstrates the importance of the proposed framework's functional modules, that is, FAIXID provides subjects with quality explanations that are overall easy to understand, efficient, and helpful in improving decision making on potential cyber threats. We observed that analysts of different roles have different expectations for the explanations provided by the attribution module. Besides, it is critical to collect analysts' feedback through the evaluation module to improve the explanation attribution. We also conducted an experimental evaluation of explainability based on proxy methods to evaluate data cleaning effectiveness for improving explainability and interpretability. The results show that the employed data cleaning mechanisms are indeed improving the interpretability score.

This work opens a path toward further enhancements and new research directions. In particular, developing an enhanced and automated attribution module capable of selecting appropriate algorithms according to the needs of analysts is foreseen as a promising new direction. Furthermore, comparing the effectiveness of different data cleaning techniques, as well as further improving the range, visual representation, and scalability of the explainability algorithms, would be an interesting future work direction. In terms of framework evaluation, it would be insightful to conduct future experiments using multiple datasets that involve a larger pool of analysts. Additionally, we plan to conduct performance analysis, using Principal Component Analysis (PCA), to figure out the relative valuable information sacrifices introduced by the data cleaning and XAI techniques. Finally, another interesting future work direction is to conduct a comprehensive comparison of the performance of the algorithms provided by the AIX360 toolkit with other algorithms in a different explainability toolkit.

Acknowledgements Awny Alnusair was supported by the IU Kokomo summer faculty fellowship program.

References

1. D'Amico, A., Whitley, K.: The real work of computer network defense analysts. In VizSEC 2007, pp 19–37. Springer, New York (2008)
2. Zhong, C., Yen, J., Liu, P., Erbacher, R.F., Garneau, C., Chen, B.: Studying analysts' data triage operations in cyber defense situational analysis. In: Theory and models for cyber situation awareness, pp. 128–169. Springer, (2017)
3. Islam, S.R., Eberle, W., Ghafoor, S.K., Siraj, A., Rogers, M.: Domain knowledge aided explainable artificial intelligence for intrusion detection and response. arXiv preprint arXiv:1911.09853, (2019)
4. Amarasinghe, K., Manic, M.: Improving user trust on deep neural networks based intrusion detection systems. In IECON 2018-44th Annual Conference of the IEEE Industrial Electronics Society, pp. 3262–3268. IEEE, (2018)
5. Chu, X., Ilyas, I.F., Krishnan, S., Wang, J.: Data cleaning: Overview and emerging challenges. In: Proceedings of the 2016 International Conference on Management of Data, pp. 2201–2206 (2016)
6. Ding, Xiaoou, Wang, Hongzhi, Su, Jiaxuan, Li, Zijue, Li, Jianzhong, Gao, Hong: Cleanits: a data cleaning system for industrial time series. Proc. VLDB Endow. **12**(12), 1786–1789 (2019)
7. Krishnan, Sanjay, Wang, Jiannan, Wu, Eugene, Franklin, Michael J, Goldberg, Ken: Activeclean: interactive data cleaning for statistical modeling. Proc. VLDB Endow. **9**(12), 948–959 (2016)
8. Yu, Z., Chu, X.: Piclean: a probabilistic and interactive data cleaning system. In: Proceedings of the 2019 International Conference on Management of Data, pp. 2021–2024 (2019)
9. Lipton, Z.C.: The mythos of model interpretability. arXiv preprint arXiv:1606.03490 (2016)
10. Onwubiko, C.: Cocoa: An ontology for cybersecurity operations centre analysis process. In: 2018 International Conference On Cyber Situational Awareness, Data Analytics And Assessment (Cyber SA), pp. 1–8. IEEE (2018)
11. Ganesan, Rajesh, Jajodia, Sushil, Shah, Ankit, Cam, Hasan: Dynamic scheduling of cybersecurity analysts for minimizing risk using reinforcement learning. ACM Trans. Intell. Syst. Technol. TIST **8**(1), 1–21 (2016)
12. Zhong, Chen, Yen, John, Liu, Peng, Erbacher, Robert F: Learning from experts' experience: toward automated cyber security data triage. IEEE Syst. J. **13**(1), 603–614 (2018)
13. Feng, C., Wu, S., Liu, N.: A user-centric machine learning framework for cyber security operations center. In: 2017 IEEE International Conference on Intelligence and Security Informatics (ISI), pp. 173–175. IEEE (2017)
14. Zhong, C., Yen, J., Liu, P.: Can cyber operations be made autonomous? an answer from the situational awareness viewpoint. In: Adaptive Autonomous Secure Cyber Systems, pp. 63–88. Springer (2020)
15. Peng, K., Leung, V.C.M., Huang, Q.: Clustering approach based on mini batch kmeans for intrusion detection system over big data. IEEE Access **6**, 11897–11906 (2018)
16. Otoum, S., Kantarci, B., Mouftah, H.: Empowering reinforcement learning on big sensed data for intrusion detection. In: IEEE International Conference on Communications (ICC), pp. 1–7 (2019)
17. Uwagbole, S.O., Buchanan, W.J., Fan, L.: Applied machine learning predictive analytics to sql injection attack detection and prevention. In: IFIP/IEEE Symposium on Integrated Network and Service Management (IM), pp. 1087–1090, Lisbon (2017)
18. Aloqaily, M., Otoum, S., Ridhawi, I.A., Jararweh, Y.: An intrusion detection system for connected vehicles in smart cities. Ad Hoc Netw. **90**, 101842. Recent advances on security and privacy in Intelligent Transportation Systems. (2019)
19. Goeschel, K.: Reducing false positives in intrusion detection systems using data-mining techniques utilizing support vector machines, decision trees, and naive bayes for off-line analysis. In: SoutheastCon 2016, pp. 1–6, Norfolk, VA (2016)
20. Hachmi, Fatma, Boujenfa, Khadouja, Limam, Mohamed: Enhancing the accuracy of intrusion detection systems by reducing the rates of false positives and false negatives through multi-objective optimization. J. Netw. Syst. Manag. **27**, 93–120 (2019)
21. Gil Pérez, Manuel, FMármol, élix Gómez, Pérez, Gregorio Martínez, Skarmeta Gómez, Antonio F.: Repcidn: a reputation-based collaborative intrusion detection network to lessen the impact of malicious alarms. J. Netw. Syst. Manag. **21**, 128–167 (2013)
22. Khosravi-Farmad, Masoud, Ghaemi-Bafghi, Abbas: Bayesian decision network-based security risk management framework. J. Netw. Syst. Manag. **28**, 1794–1819 (2020)

23. Otoum, S., Kantarci, B., Mouftah, H.T.: Mitigating false negative intruder decisions in wsn-based smart grid monitoring. In: 13th International Wireless Communications and Mobile Computing Conference (IWCMC), pp. 153–158 (2017)
24. Liu, Y., Sarabi, A., Zhang, J., Naghizadeh, P., Karir, M., Bailey, M., Liu, M.: Cloudy with a chance of breach: Forecasting cyber security incidents. In: 24th USENIX Security Symposium (USENIX Security 15), pp. 1009–1024, Washington, D.C., (August 2015). USENIX Association
25. Soska, K., Christin, N.: Automatically detecting vulnerable websites before they turn malicious. In: 23rd USENIX Security Symposium (USENIX Security 14), pp. 625–640, San Diego, CA, (August 2014). USENIX Association
26. Ritter, A., Wright, E., Casey, W.A., Michael, T.: Weakly supervised extraction of computer security events from twitter. In: Proceedings of the 24th International Conference on World Wide Web WWW, pp. 896–905 (2015)
27. Yang, H., Ma, X., Du, K., Li, Z., Duan, H., Su, X., Liu, G., Geng, Z., Wu, J.: How to learn klingon without a dictionary: Detection and measurement of black keywords used by the underground economy. In: 2017 IEEE Symposium on Security and Privacy (SP), pp. 751–769 (2017)
28. Sabottke, Carl., Suciu, Octavian., Dumitras, Tudor.: Vulnerability disclosure in the age of social media: Exploiting twitter for predicting real-world exploits. In 24th USENIX Security Symposium (USENIX Security 15), pages 1041–1056, Washington, D.C., USENIX Association. (2015)
29. Borgolte, K., Kruegel, C., Vigna, G.: Delta: automatic identification of unknown web-based infection campaigns. In: The ACM SIGSAC Conference on Computer and Communications Security, pp. 109–120 (2013)
30. Wang, M., Zheng, K., Yang, Y., Wang, X.: An explainable machine learning framework for intrusion detection systems. IEEE Access **8**, 73127–73141 (2020)
31. Chandrasekaran, B., Tanner, M.C., Josephson, J.R.: Explaining control strategies in problem solving. IEEE Intell. Syst. (1), 9–15 (1989)
32. Swartout, W.R., Moore, J.D.: Explanation in second generation expert systems. In: Second Generation Expert Systems, pp. 543–585. Springer (1993)
33. Swartout, W.R.: Rule-based expert systems: The mycin experiments of the stanford heuristic programming project: Bg buchanan and eh shortliffe,(Addison-Wesley, Reading, MA, 1984), p. 702 (1985)
34. Esper, M.T.: Ai ethical principles. https://www.defense.gov/Newsroom/Releases/Release/Article/2091996/dod-adopts-ethical-principles-for-artificial-intelligence/, (February 2020). Accessed 03 July 2020
35. Yang, S.C.-H., Shafto, P.: Explainable artificial intelligence via bayesian teaching. In: NIPS 2017 Workshop on Teaching Machines, Robots, and Humans (2017)
36. Lei, T., Barzilay, R., Jaakkola, T.: Rationalizing neural predictions. arXiv preprint arXiv:1606.04155 (2016)
37. Ribeiro, M.T., Singh, S., Guestrin, C.: 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, pp. 1135–1144. ACM (2016)
38. Kim, B., Wattenberg, M., Gilmer, J., Cai, C., Wexler, J., Viegas, F., Sayres, R.: Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). arXiv preprint arXiv:1711.11279 (2017)
39. Horel, E., Giesecke, K.: Towards explainable ai: Significance tests for neural networks. arXiv preprint arXiv:1902.06021 (2019)
40. Marino, D.L., Wickramasinghe, C.S., Manic, M.: An adversarial approach for explainable ai in intrusion detection systems. In: IECON 2018—44th Annual Conference of the IEEE Industrial Electronics Society, pp. 3237–3243 (2018)
41. Hartl, A., Bachl, M., Fabini, J., Zseby, T.: Explainability and adversarial robustness for rnns. In: 2020 IEEE Sixth International Conference on Big Data Computing Service and Applications (BigDataService), pp. 148–156 (2020)
42. Kasun Amarasinghe, K., Kenney, K., Manic, M.: Toward explainable deep neural network based anomaly detection. In: 2018 11th International Conference on Human System Interaction (HSI), pp. 311–317 (2018)
43. Wang, Zhidong, Lai, Yingxu, Liu, Zenghui, Liu, Jing: Explaining the attributes of a deep learning based intrusion detection system for industrial control networks. Sensors **20**(14), 3817 (2020)
44. Al Ridhawi, Ismaeel, Otoum, Safa, Aloqaily, Moayad, Boukerche, Azzedine: Generalizing AI: challenges and opportunities for plug and play AI solutions. IEEE Netw. **35**(1), 372–379 (2020)

45. Lundberg, S.M., Lee, S.I.: A unified approach to interpreting model predictions. In: Advances in Neural Information Processing Systems, pp. 4765–4774 (2017)
46. Shrikumar, A., Greenside, P., Kundaje, A.: Learning important features through propagating activation differences. In: Proceedings of the 34th International Conference on Machine Learning, Vol. 70, pp. 3145–3153. JMLR.org (2017)
47. Ando, S.: Interpreting random forests. http://blog.datadive.net/interpreting-random-forests/ (2019)
48. Datta, A., Sen, S., Zick, Y.: Algorithmic transparency via quantitative input influence: Theory and experiments with learning systems. In: 2016 IEEE Symposium on Security and Privacy (SP), pp. 598–617. IEEE (2016)
49. Štrumbelj, Erik, Kononenko, Igor: Explaining prediction models and individual predictions with feature contributions. Knowl. Inf. Syst. **41**(3), 647–665 (2014)
50. Lipovetsky, Stan, Conklin, Michael: Analysis of regression in game theory approach. Appl. Stoch. Models Bus. Ind. **17**(4), 319–330 (2001)
51. Bach, Sebastian, Binder, Alexander, Montavon, Grégoire, Klauschen, Frederick, Müller, Klaus-Robert, Samek, Wojciech: On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. PLoS ONE **10**(7), e0130140 (2015)
52. Lundberg, Scott.: Shap vs lime
53. 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.: Explainable artificial intelligence (xai): Concepts, taxonomies, opportunities and challenges toward responsible ai. Inf. Fusion **58**, 82–115 (2020)
54. Liu, H., Kim, J.: Data quality assessment and problem severity assessment for data cleaning. In: The 15th International Conference on Data Science, pp. 207–210 (2019)
55. Dash, S., Gunluk, O., Wei, D.: Boolean decision rules via column generation. In: Advances in Neural Information Processing Systems, pp. 4655–4665 (2018)
56. Islam, S.R., Eberle, W., Bundy, S., Ghafoor, S.K.: Infusing domain knowledge in ai-based" black box" models for better explainability with application in bankruptcy prediction. arXiv preprint arXiv:1905.11474 (2019)
57. Dhurandhar, A., Chen, P.-Y., Luss, R., Tu, C.-C., Ting, P., Shanmugam, K., Das, P.: Explanations based on the missing: towards contrastive explanations with pertinent negatives. In: Advances in Neural Information Processing Systems, pp. 592–603 (2018)
58. Zhang, C., Bengio, S., Hardt, M., Recht, B., Vinyals, O.: Understanding deep learning requires rethinking generalization. arXiv preprint arXiv:1611.03530 (2016)
59. Doshi-Velez, F., Kim, B.: Towards a rigorous science of interpretable machine learning. arXiv preprint arXiv:1702.08608 (2017)
60. Melis, D.A., Jaakkola, T.: Towards robust interpretability with self-explaining neural networks. In: Advances in Neural Information Processing Systems, pp. 7775–7784 (2018)
61. Luss, R., Chen, P.-Y., Dhurandhar, A., Sattigeri, P., Zhang, Y., Shanmugam, K., Tu, C.-C.: Generating contrastive explanations with monotonic attribute functions. arXiv preprint arXiv:1905.12698 (2019)
62. Maciá-Fernández, Gabriel, Camacho, José, Magán-Carrión, Roberto, García-Teodoro, Pedro, Therón, Roberto: Ugr'16: a new dataset for the evaluation of cyclostationarity-based network idss. Comput. Secur. **73**, 411–424 (2018)
63. Arya, V., Bellamy, R. K.E., Chen, P.-Y., Dhurandhar, A., Hind, M., Hoffman, S.C., Houde, S., Liao, Q.V., Luss, R., Mojsilović, A. et al.: One explanation does not fit all: A toolkit and taxonomy of ai explainability techniques. arXiv preprint arXiv:1909.03012 (2019)
64. Wei, D., Dash, S., Gao, T., Günlük, O.: Generalized linear rule models. arXiv preprint arXiv:1906.01761 (2019)
65. Gurumoorthy, K.S., Dhurandhar, A., Cecchi, G., Aggarwal, C.: Efficient data representation by selecting prototypes with importance weights. In: 2019 IEEE International Conference on Data Mining (ICDM), pp. 260–269. IEEE (2019)
66. Islam, S.R., Eberle, W., Ghafoor, S.K.: Towards quantification of explainability in explainable artificial intelligence methods. AAAI Publications, The Thirty-Third International Flairs Conference (2020)
67. Miller, George A: The magical number seven, plus or minus two: some limits on our capacity for processing information. Psychol. Rev. **63**(2), 81 (1956)
68. Wolf, C.T., Ringland, K.E.: Designing accessible, explainable ai (xai) experiences. In: ACM SIGACCESS Accessibility and Computing (125):1–1 (2020)

Hong Liu received her Ph.D. degree in Computer Science from Oklahoma State University. She is an Assistant Pofessor of Computer Science at Indiana University Kokomo. Her research interests include Data Cleaning, Data Quality Management, Artificial Intelligence, and Cybersecurity.

Chen Zhong is an Assistant Professor of Cybersecurity in the Information and Technology Management Department at the University of Tampa. She received a Ph.D. degree in Information Sciences and Technology from the Pennsylvania State University. Her research interests include Intrusion Detection, Cybersecurity Situation Awareness, and Artificial Intelligence.

Awny Alnusair received a Ph.D. degree in Computer Science from the University of Wisconsin–Milwaukee. He is currently an Associate Professor of Informatics and Computer Science at Indiana University Kokomo. His research interests include Software Engineering, Data Mining in VANETs, Cloud Computing, Big Data Analytics, and Cybersecurity.

Sheikh Rabiul Islam is an Assistant Professor of Computer Science at the University of Hartford. His notable research experiences are in the area of Explainable Artificial Intelligence (XAI), Data Mining and Big Data Analytics, and Cybersecurity. He is a member of AAAI and IEEE.

Authors and Affiliations

Hong Liu[1] · Chen Zhong[2] · Awny Alnusair[1] · Sheikh Rabiul Islam[3]

✉ Awny Alnusair
alnusair@iu.edu

Hong Liu
hlius@iu.edu

Chen Zhong
czhong@ut.edu

Sheikh Rabiul Islam
shislam@hartford.edu

[1] Indiana University Kokomo, 2300 South Washington St, Kokomo, IN 46904, USA

[2] University of Tampa, 401 W Kennedy Blvd, Tampa, FL 33606, USA

[3] University of Hartford, 200 Bloomfield Ave, West Hartford, CT 06117, USA