A Survey on Explainable Artificial Intelligence for Internet Traffic Classification and Prediction, and Intrusion Detection

Alfredo Nascita, Giuseppe Aceto, Domenico Ciuonzo, *Senior Member, IEEE*,
Antonio Montieri, Valerio Persico, and Antonio Pescapé, *Senior Member, IEEE*

***Abstract*—With the increasing complexity and scale of modern networks, the demand for transparent and interpretable Artificial Intelligence (AI) models has surged. This survey comprehensively reviews the current state of eXplainable Artificial Intelligence (XAI) methodologies in the context of Network Traffic Analysis (NTA) (including tasks such as traffic classification, intrusion detection, attack classification, and traffic prediction), encompassing various aspects such as techniques, applications, requirements, challenges, and ongoing projects. It explores the vital role of XAI in enhancing network security, performance optimization, and reliability. Additionally, this survey underscores the importance of understanding why AI-driven decisions are made, emphasizing the need for explainability in critical network environments. By providing a holistic perspective on XAI for Internet NTA, this survey aims to guide researchers and practitioners in harnessing the potential of transparent AI models to address the intricate challenges of modern network management and security.**

***Index Terms*—explainable artificial intelligence; network traffic analysis; Internet traffic classification; Internet traffic prediction; intrusion detection; deep learning.**

I. Introduction

The explainability of Artificial Intelligence (AI) has become centerfold in the field of Internet traffic analysis and monitoring (aiming at traffic management and security). Indeed, in response to the formidable requirements of modern Internet traffic, recent research has placed a significant emphasis on the utilization of Machine Learning (ML) and Deep Learning (DL) techniques [1] to create efficient tools for addressing Internet traffic classification (targeting either legit apps/services or attacks), intrusion detection, and Internet traffic prediction, hereafter collectively referred as *Network Traffic Analysis (NTA)* tasks. While these methods hold the promise of delivering exceptional performance and the ability to autonomously adapt to evolving traffic patterns, they inherently function as *black-box* systems, making it exceedingly challenging to comprehend, enhance, or safeguard their behavior against potential attacks. Consequently, trust in these methods is limited.

In fact, wrap-up performance metrics (e.g., accuracy) are no longer sufficient in contexts as critical as network management; trust in AI algorithms is essential for network administrators and users to make informed decisions and determine appropriate actions. This need is even more pronounced as AI is increasingly proposed to handle the complexities of real-time, evolving, large-scale, and distributed network resource management, progressively leaving the human out of the loop [2]. Hence, there is a unanimous consensus among key stakeholders that there is a pressing need for explainability of AI solutions in the broad context of networking. Explanations for AI systems have evolved from being optional features to becoming the cornerstone of any AI design solution that users and network operators can deem safe, dependable, controllable, and equitable.

As a consequence of the many and heterogeneous needs for explainability or interpretability, there is no lack of definitions for these properties [3]. Therefore, we apply one of the most comprehensive definitions for eXplainable Artificial Intelligence (XAI), based on the *explanation*: “An explanation is a presentation of (aspects of) the reasoning, functioning and/or behavior of a ML model in human-understandable terms” [4]. Here *reasoning* refers to the process leading from inputs to outputs, *functioning* is related to the structural characteristics (data representation), and *behavior* describes the input-output relationships regardless of the internals. The three aspects are non-exclusive: an explanation may participate in more than one. Nauta et al. [4] conflate explainability with interpretability and intelligibility, stating that there is no agreement on the differences among the three. We share this view, and add that often transparency is also used to refer to explainability, in contrast with the “black-box” metaphor.

A. Contribution and Survey Organization

This article contributes to the burgeoning field of XAI within the context of NTA. More specifically, our contributions are listed in the following, with reference to the structure of the paper.

- We discuss in Section II the **motivation** behind our literature analysis, defining the context and the related studies, and highlighting the **shared interest in XAI as expressed by governmental and telecommunication stakeholders**, as well as the gap in the scientific literature we aim to fill with our study.
- We introduce in Section III a **practical taxonomy of XAI methods**, providing essential background to support the

Manuscript received 7th May 2024; revised 13th October 2024; accepted 17th November 2024.

The authors are with the Department of Electrical Engineering and Information Technologies (DIETI) at the University of Naples Federico II, Italy.

E-mail: {alfredo.nascita, giuseppe.aceto, domenico.ciuonzo, antonio.montieri, valerio.persico, pescape}@unina.it.

The illustrations in this manuscript include images downloaded from *flaticon.com* and *fontawesome.com*.

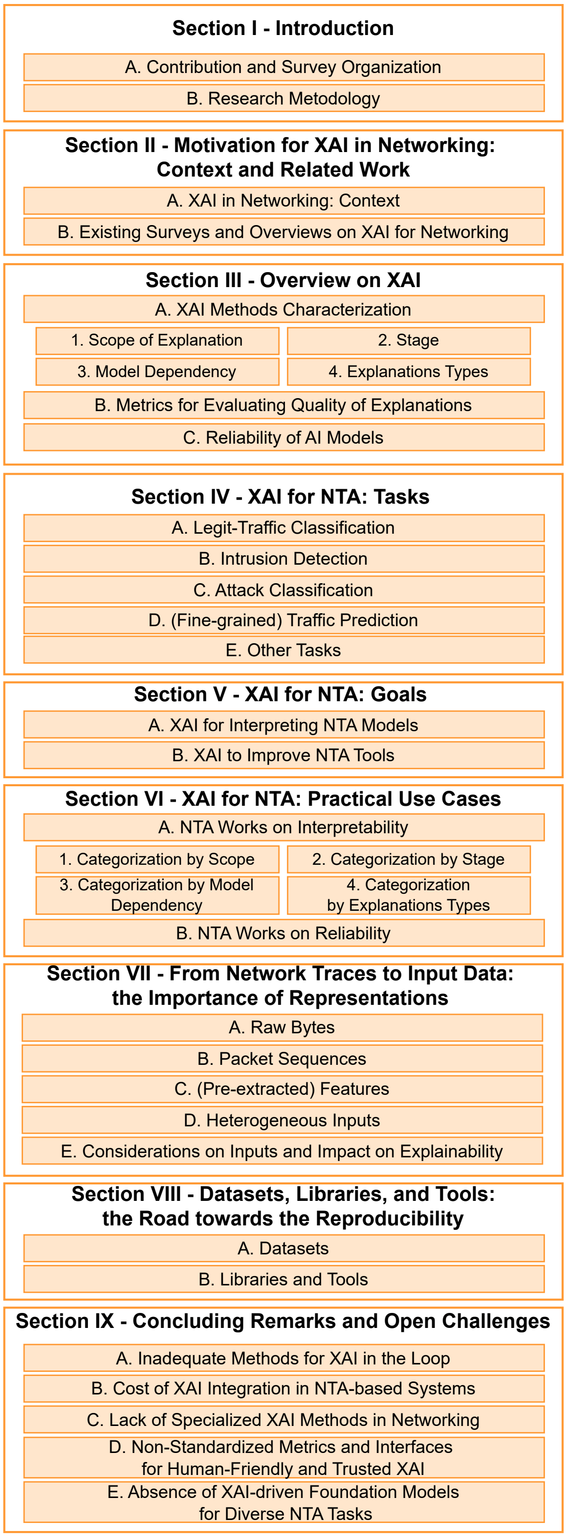

Section I - Introduction
A. Contribution and Survey Organization
B. Research Metodology
Section II - Motivation for XAI in Networking: Context and Related Work
A. XAI in Networking: Context
B. Existing Surveys and Overviews on XAI for Networking
Section III - Overview on XAI
A. XAI Methods Characterization
1. Scope of Explanation
2. Stage
3. Model Dependency
4. Explanations Types
B. Metrics for Evaluating Quality of Explanations
C. Reliability of AI Models
Section IV - XAI for NTA: Tasks
A. Legit-Traffic Classification
B. Intrusion Detection
C. Attack Classification
D. (Fine-grained) Traffic Prediction
E. Other Tasks
Section V - XAI for NTA: Goals
A. XAI for Interpreting NTA Models
B. XAI to Improve NTA Tools
Section VI - XAI for NTA: Practical Use Cases
A. NTA Works on Interpretability
1. Categorization by Scope
2. Categorization by Stage
3. Categorization by Model Dependency
4. Categorization by Explanations Types
B. NTA Works on Reliability
Section VII - From Network Traces to Input Data: the Importance of Representations
A. Raw Bytes
B. Packet Sequences
C. (Pre-extracted) Features
D. Heterogeneous Inputs
E. Considerations on Inputs and Impact on Explainability
Section VIII - Datasets, Libraries, and Tools: the Road towards the Reproducibility
A. Datasets
B. Libraries and Tools
Section IX - Concluding Remarks and Open Challenges
A. Inadequate Methods for XAI in the Loop
B. Cost of XAI Integration in NTA-based Systems
C. Lack of Specialized XAI Methods in Networking
D. Non-Standardized Metrics and Interfaces for Human-Friendly and Trusted XAI
E. Absence of XAI-driven Foundation Models for Diverse NTA Tasks

Figure 1. Survey organization.

reader in understanding the peculiarities of NTA-specific research efforts and applications.

- We shed light on how the adoption of XAI proved fruitful in Section IV, highlighting the **4 recurrent tasks in the NTA domain XAI is applied to** and focusing on the interest trends in the last few years.
- We analyze in depth the **purposes driving the adoption of XAI in NTA** in Section V, highlighting how such techniques are useful both for the interpretation and the improvement of black-box AI models.
- In the same Section, we provide a “translation table” with the heterogeneous and non-standardized **explanation-quality metrics as adopted in NTA**, mapping them to the systematized categories proposed in the broader XAI study field (and highlighting the missing viewpoints); for each metric, we also provide its formal definition, along with the indication of its applicability to different XAI methods and related NTA tasks.
- We categorize and extensively discuss the **practical use cases of XAI for NTA** in Section VI, leveraging the taxonomy we proposed.
- Since network traffic can be subjected to different representations, we discuss in Section VII **how the representation of the input data influences the interpretability** of the models, underlying the peculiarities of NTA domain with respect to those where XAI techniques have been initially developed (e.g., computer vision).
- Being reproducibility a main concern in the NTA field, we survey existing **public datasets** in Section VIII, and report their usage (or lack thereof) in XAI analyses, thus providing the interested reader with valuable and ready-to-use resources.
- In the same section, we scout for **open source libraries and tools** highlighting those that provide explanation-quality metrics.
- Finally, we draw the conclusions of the survey and discuss **open challenges and gaps** arising from the application of XAI to NTA in Section IX.

Figure 1 outlines the organization of the present survey sketching the details of the sections that constitute the manuscript. Furthermore, Table I contains the acronyms and abbreviations defined in the present paper to aid readability.

B. Research Methodology

To conduct our study, we have adopted a systematic approach to exploring the literature, following the methodology outlined by Wohlin [5]. For the aims and scope of our paper, we composed the following query:

> *(“explainable AI” **OR** “interpretable AI”)*
> ***AND** (“network traffic analysis”)*

The Google Scholar search engine has been selected, to avoid bias on publishers according to best practices. From the obtained results, we initially selected a set of 28 papers for the snowballing methodology. Our selection criteria included papers written in English with full-text accessibility either publicly or through a subscription by the University of Napoli Federico II. We considered the relevance of the title and the number of citations to prioritize papers for inclusion. To carry out the snowballing process, we employed both backward and forward methods. During this process, we evaluated the abstracts for relevance and, in the case of potential candidates

Table I
LIST OF ACRONYMS AND ABBREVIATIONS IN ALPHABETIC ORDER.

Acronym	Definition
AC	Attack Classification
AD	Anomaly Detection
AE	AutoEncoder
AI	Artificial Intelligence
BRCG	Boolean Decision Rules via Column Generation
CADE	Contrastive Autoencoder for Drifting detection and Explanation
CEM	Contrastive Explanation Method
CNN	Convolutional Neural Network
COIN	Contextual Outlier INterpretation
DARPA	Defense Advanced Research Projects Agency
DDos	Distributed Denial of Service
DeepLIFT	Deep Learning Important FeaTures
DIR	Direction
DL	Deep Learning
DNN	Deep Neural Network
DT	Decision Tree
EDA	Exploratory Data Analysis
FL	Focal Loss
FPGA	Field Programmable Gate Array
Grad-CAM	Gradient-weighted Class Activation Mapping
IAT	Inter Arrival Time
ICE	Individual Conditional Expectation
ID	Intrusion Detection
IDS	Intrusion Detection System
IG	Integrated Gradients
IoT	Internet of Things
KNN	K-Nearest Neighbors
LEMNA	Local Explanation Method using Nonlinear Approximation
LIME	Local Interpretable Model-agnostic Explanations
LRP	Layer-wise Relevance Propagation
LS	Label Smoothing
LSTM	Long Short-Term Memory
MD	Misuse Detection
ML	Machine Learning
NTA	Network Traffic Analysis
PDP	Partial Dependence Plot
PFI	Permutation Feature Importance
PL	Payload Length
PS	Packet Size
QoS	Quality of Service
SDN	Software Defined Networking
SHAP	SHapley Additive exPlanations
SNI	Server Name Indication
SOM	Self Organizing Map
SVM	Support Vector Machine
TC	Traffic Classification
TCP WS	TCP Window Size
TLS	Transport Layer Security
TO	Traffic Object
TP	Traffic Prediction
t-SNE	t-distributed Stochastic Neighbor Embedding
TTL	Time To Live
XAI	eXplainable Artificial Intelligence

for inclusion, we also reviewed the full papers. As a result of this thorough process, we obtained a comprehensive set of 107 papers.

II. MOTIVATION FOR XAI IN NETWORKING: CONTEXT AND RELATED WORK

Our work is strongly motivated by the absence of comprehensive studies that cover the adoption of XAI for NTA despite the great consideration for XAI in networking shown by the stakeholders at diverse layers. In Section II-A, we discuss such interest witnessed by the principles and guidelines as expressed by regulators and telecommunication companies. Then, in Section II-B, we review the related literature, highlighting the substantial lack of surveys about the adoption of XAI in the domain of NTA.

A. XAI in Networking: Context

The shared commitment to building a responsible and sustainable AI ecosystem in networking is reflected in efforts by *private stakeholders* and *regulators*. As technology evolves, these principles guide the research, deployment, and management of AI-driven networking, promoting reliable, secure infrastructures. Key efforts by governmental agencies and telecommunication companies are summarized in Table II and discussed below.

Defense Advanced Research Projects Agency (DARPA) launched the XAI program in 2017 [9] to create more explainable AI models, design effective interfaces, and understand the psychological needs for explanations. The EU AI Act [11], finalized in January 2024, restricts AI systems that affect lives unless they ensure transparency or explainability, especially in high-risk areas. This aligns with the outcomes of EU High-Level Expert Group in 2018, where transparency was already identified as a key requirement for trustworthy AI [6]. The 2023 White House executive order also highlights the need for AI model transparency for regulated entities [15].

These requirements are especially relevant in communications and networking, where telecommunications companies are developing AI-driven solutions as secure foundations for their core business operations. Telefonica's "Responsible Use of AI" [8] addresses discrimination, interpretability, and data transparency, focusing on *fair, transparent/explainable, and human-centric* AI as well as privacy and security by design. Numerous other organizations and firms have established guidelines and principles to direct their research and servicing endeavors. In line with the EU Commission's guidelines for *Trustworthy AI*, Ericsson elaborated on this concept [10], primarily concentrating on the explainability, safety, and verifiability of AI solutions. They have also incorporated aspects like traceability and accountability, consistently placing the human being at the core of the entire process. Nokia's AI pillars [14] include transparency to ensure trustworthiness and help engineers improve model accuracy. Explainability and transparency also aid in root cause analysis, especially for troubleshooting. In its "Principles for Responsible AI" [16], Cisco also mentions transparency, intended as communicating with the users "when and how AI is employed". Similarly, Juniper's AI innovation principles include transparency [13] meaning clarity about the adoption of AI in products. However, Cisco adds to its principles *reliability* meaning "how reliably

Table II
Principles and Guidelines related to XAI in (networking) systems presented in chronological order (↓).

Issuing Entity	Description	Year↓	Ref.
EU⚖	Ethics guidelines for trustworthy AI	2018[†]	[6]
Huawei	AI Security White Paper	2018	[7]
Telefonica	Telefonica's Approach to the Responsible Use of AI	2018	[8]
DARPA⚖	DARPA's eXplainable Artificial Intelligence (XAI) Program	2019	[9]
Ericsson	Trustworthy AI: explainability, safety and verifiability	2020	[10]
EU⚖	EU AI Act: first regulation on artificial intelligence	2021[*]	[11]
NEC	NEC AI Guide Book	2021	[12]
Juniper	Explainable AI explained	2023	[13]
Nokia	Responsible AI for telecom	2023	[14]
White House⚖	Executive Order on the Safe, Secure, and Trustworthy Development and Use of AI	2023	[15]
Cisco	Cisco Principles for Responsible AI	2024	[16]

⚖ The issuing entity is a regulator or government body/agency.
[†] EU High-Level Expert Group on AI in 2018, final version in 2019 after open consultation.
[*] EU Commission Draft in 2021, EU Parliament approved Final Draft in 2024.

that solution produces a desired output based on the data set on which it has been (continuously) trained", and it also aims at fairness (by identifying and remediating any harmful AI bias). While these properties are possibly related to explainability, they do not strictly match with explainability as pursued in the XAI study field. On the other hand, Juniper also explicitly considers *explainability* "with a goal toward having explainable decision-making processes and intended impact". Huawei identifies the lack of explainability as the main cause of security vulnerabilities in AI systems [7]. This gap creates unique security risks in DL-based applications, making them vulnerable to adversarial ML attacks like evasion, poisoning, and backdoor exploits. Finally, NEC allows black-box AI for efficiency improvements but emphasizes the need for XAI in scenarios beyond efficiency [12].

While XAI is a challenging research in most AI applications, in networking there are additional specific difficulties, mostly ascribable—at least partially—to the complex and highly-structured nature of input data, deriving from *traffic traces* captured by a network probe. As DL is relatively new in NTA, XAI applications are still maturing, though its need is clear, as evidenced by a growing body of literature. In line with such an increasing interest, a number of studies surveyed the existing literature on XAI in networking-related domains. We discuss them in the following subsection in order to clearly position our contribution.

B. Existing Surveys and Overviews on XAI for Networking

In recent years, an expanding corpus of surveys has explored the realm of XAI within a *variety of application domains*, encompassing, for instance, natural language processing [30], healthcare [31], and energy sectors [32]. This body of literature witnesses the interest and engagement of the broader research community in this area and the *key need for contextualization of explainability issues and desiderata in each peculiar domain*. Notably, within the large field of *networking*, there has been a proliferation of surveys that delve into the realm of XAI as highlighted in Table III, in which we pinpoint the *networking domains* considered in such related surveys (column "**Domain**"). Nevertheless, existing survey efforts comprehensively dissecting NTA, mostly focus on aspects such as novel opaque ML/DL design methodologies [33, 34], and XAI is overlooked or not even mentioned as a future direction. This trend underscores the paramount importance of enhancing the transparency, interpretability, and comprehensibility of NTA. As XAI techniques mature, integrating them into NTA promises to bolster network resilience and security. The growing research in this area acknowledges XAI's pivotal role in addressing modern NTA challenges, offering promising implications for both academia and industry. Zhang et al. [20] provide a general overview of the landscape in AI-driven networking solutions and systems, suggesting forthcoming challenges and prospective trajectories, with a specific focus on *network management*. The authors suggest the potential of XAI to elevate AI-based networking solutions and provide an initial discussion of different *XAI goals* such as performance, feasibility, robustness, and trust. This article is proposed as a primitive guidance for the incremental improvement of AI-based networking solutions, and as an opportunity to foster a discussion on the necessity of XAI in networking.

Other surveys focus on specific application domains such as network security, the Internet of Things (IoT) paradigm, or the emerging applications of 5G networks and beyond. Differently, *our study focuses on network traffic*, which is the core information processed by AI for all these verticals. Hence, the adoption of XAI in the NTA domain overlaps with all these above. However, the juxtaposition of the contributions of these surveys—besides not being trivial—does not provide a comprehensive or consistent view of the peculiarities of XAI in NTA. Our longstanding research experience on NTA, AI, and XAI supports a thorough investigation that both allows understanding the specific application domains and surfacing the common solutions and open problems. Indeed, **our contributions include an in-depth background on XAI**

Table III
EXISTING OVERVIEWS AND SURVEYS ON XAI IN NETWORKING-RELATED DOMAINS. WORKS ARE LISTED IN CHRONOLOGICAL ORDER (↓).

		Domain				Focus						
Venue [Ref.]	**Year ↓**	**Netw. Manag.**	**Security**	**IoT**	**(B)5G**	**Background on XAI Methods**	**XAI Goals**	**Metrics**	**Extensive Discussion of Use Cases**	**Input**	**Reproducibility**	**Open Challenges**
IEEE ISEA-ISAP [17]	2021		✓			◐	○	○	○	○	●	○
IEEE Open J. Commun. Soc. [18]	2022			✓	✓	●	◐	○	●	○	○	●
IEEE Access [19]	2022		✓			●	○	○	●	○	◐	●
IEEE Commun. Mag. [20]	2022	✓				○	◐	○	○	○	○	●
IEEE Access [21]	2022		✓			◐	○	○	●	○	○	●
IEEE Access [22]	2022		✓			●	○	◐	●	○	◐	●
IEEE BigData [23]	2022		✓			○	○	◐	○	○	○	○
Comput. Commun. [24]	2022				✓	◐	○	○	○	○	○	○
IEEE Trans. Netw. Service Manag. [25]	2023		✓			●	○	○	◐	○	○	●
IEEE Internet Things J. [26]	2023			✓		●	◐	○	●	◐	◐	●
IEEE Commun. Surv. Tutor. [27]	2023		✓	✓		●	○	○	○	○	○	●
IEEE Open J. Commun. Soc. [28]	2024				✓	●	●	○	●	○	○	●
IEEE Commun. Surv. Tutor. [29]	2024		✓		✓	●	◐	○	◐	○	○	●
This Survey	2024	✓	✓	✓		●	●	●	●	●	●	●

methods, the analysis of the NTA tasks for which XAI is leveraged, its goals, the most relevant practical use cases, the input data exploited to feed the models, the relevant public datasets and tools for reproducibility purpose, as well as the analysis of the open challenges. To summarize the above discussion, in Table III, we present an overview of the main contributions (column "**Focus**") of the present survey (last row) compared to related ones. Going into more detail, hereafter, we discuss the relevant surveys on the adoption of XAI organized by the specific domain they focus on.

Security: Unquestionably, the majority of scholarly surveys are centered around the domain of *(cyber)security and its diverse applications*. For instance, Moustafa et al. [27] examine the value of explainability techniques focusing on the intersection of XAI, Intrusion Detection Systems (IDSs), and IoT. They remark that explaining ML and DL models for cyber defense systems is vital and highlight the lack of standardized terminology and evaluation of explanation. Capuano et al. [21] delve into critical facets of cybersecurity applications, encompassing areas like IDSs, malware detection, phishing and spam detection, botnets detection, fraud detection, zero-day vulnerabilities, digital forensics, and crypto-jacking. This work primarily focuses on evaluating the employment of explainability techniques within these domains, emphasizing notable accomplishments and emerging challenges. It also engages in a comprehensive discussion of prior research, ongoing trends, and forthcoming obstacles in this context. In contrast, Zhang et al. [19] conduct an assessment of the integration of cybersecurity measures across a diverse array of industrial sectors, which includes the domains of smart grid, healthcare, smart agriculture, smart transportation, human-computer interaction, and smart financial systems. This survey also ventures into the realm of adversarial threats aimed at XAI models, thoroughly examining both the nature of these attacks and the defensive strategies. In another relevant work, Rjoub et al. [25] introduce general principles and methodologies derived from the XAI literature while analyzing their potential advantages when applied in the domain of cybersecurity. They provide a comprehensive classification of the cybersecurity field, acting as a framework to explore the literature and potential applications of XAI in sub-domains like privacy, investigation, access control, intrusion detection and prevention, trust, and reputation. Furthermore, this paper delves into future research prospects and ethical concerns related to XAI in this context. Moreover, Hariharan et al. [17] strive to offer insights into the terminology, classification, and scope pertaining to XAI within the realm of cybersecurity. This work also examines the challenges encountered and the evaluation metrics applied, drawing valuable insights from existing literature. The review process encompasses four distinct stages, including exploratory data analysis, XAI methods, the availability of XAI toolkits, and the critical phase of evaluating explanations. The survey by Neupane et al. [22] conducts an extensive investigation of the current state of XAI within the specific context of IDSs. This work evaluates the main challenges in the field and examines their impact on the development of eXplainable IDSs (X-IDSs). The survey provides a comprehensive examination of both black-box and white-box approaches, scrutinizing the trade-offs concerning performance and their capacity to generate explanations. Moreover, the authors propose a generic architecture that integrates human-in-the-loop components, serving as a valuable reference for designing X-IDS. In contrast, Warmsley et al. [23] concentrate on the interpretability of graph neural networks as applied to malware detection. Their study involves a comparative analysis of various methods using diverse metrics to assess the explanations and the time required for execution. As highlighted in Table III, most surveys in the security domain cover both the background of XAI and relevant open issues (with the exception of [23], which lacks both). However, they often fail to discuss metrics and input data, or do not provide sufficient details on usable datasets and tools, which is a significant shortcoming given the unique challenges of the security domain.

IoT: In the *realm of the IoT*, an analysis of XAI frameworks is presented by Jagatheesaperumal et al. [18], examining their characteristics and evaluating their suitability. It illustrates the common adoption of XAI in various IoT applications, like the Internet of Medical Things, Industrial IoT, and Internet of City Things. Furthermore, this work delves into the most recent advancements in XAI-based architectures and their integration into 6G communication services tailored for IoT applications. Additionally, Kök et al. [26] offer an extensive and systematic review of recent research employing XAI models. The surveyed studies are categorized based on their methodology and application domains, which include autonomous systems and robotics, energy management, environmental monitoring, financial systems, healthcare, industrial domain, security and privacy, and smart agriculture. Table III underlines that all surveys in the IoT domain emphasize the discussion on open challenges, while also providing in-depth background on XAI, typically tailored to the specific application context. Notably, the survey in [27] is the only one that does not extensively discuss relevant IoT use cases for XAI which is a key pillar of the other works.

(B)5G: Other surveys are framed in the context of *(beyond-)5G applications*. Senevirathna et al. [29] explore XAI within the realm of security, encompassing technical aspects, applications, prerequisites, constraints, challenges, ongoing projects, standardization efforts, and valuable insights, all tailored to the context of beyond-5G applications. Wang et al. [28] survey the application of XAI in the context of the upcoming 6G era. The work encompasses various aspects, including 6G technologies like intelligent radio and zero-touch network management, as well as 6G use cases such as Industry 5.0. This survey summarizes insights from recent endeavors and outlines critical research challenges for the prospective integration of XAI into 6G networks. Fiandrino et al. [24] review the current status of tools and methods for enhancing the robustness and explainability of AI, listing challenges, open issues, and potential future research directions. The authors examine strategies for enabling robust and interpretable AI within 6G networks and its incorporation into existing network architecture models. They explore a case study to demonstrate how XAI tools can provide explanations linked to the characteristics of input data. Additionally, this paper investigates the computational demands involved in executing XAI tools, including the presentation of execution time and resource utilization metrics. Similar to the surveys in the IoT domain, the works here also pay particular attention to XAI background and its application to (B)5G use cases, as well as related open issues (except for the study in [24], which partially overlooks both). Unfortunately, none of the (B)5G-related surveys consider metrics, input data, or research reproducibility aspects, which we believe are particularly important in this challenging and rapidly evolving domain.

III. Overview on XAI

To cover the range of concepts and techniques that contribute to make AI "eXplainable", we first provide a characterization of XAI methods in Section III-A; then, we analyze the metrics for quantitatively assessing the quality of the explanations in Section III-B; finally, we discuss the reliability of the AI models in Section III-C.

A. XAI Methods Characterization

XAI methods can be characterized along different dimensions, as depicted in Figure 2. These dimensions include the scope of the explanations they offer, the model training phase at which they are applied, their reliance on the model being interpreted, and the type of explanations provided. This discussion can help interested readers understand and identify the characteristics of the XAI method that best suit the explainability analysis they intend to carry out.

1) Scope of Explanation — Local vs. Global: **Local XAI** methods concentrate on elucidating the decision-making process for individual predictions generated by a model [35]. They shed light on *why a specific decision was reached for a particular instance of input data*. For example, in network traffic analysis, a local XAI method can explain why a model flagged a specific network flow as malicious, thus allowing detailed verification of the model's behavior for each instance of interest. In contrast, **global XAI methods** aim to impart a more comprehensive understanding of the overall behavior and operation of the model [36]. They achieve this by unveiling patterns, rules, and relationships across the entire dataset or a subset of it, thus providing a holistic perspective on the model's behavior. A global XAI method can shed light on the overall behavior of a system that determines the traffic class to which a packet flow belongs, providing insights into the characteristics that guide the whole classification process. It is worth mentioning that there are various approaches to transition from local explanations to global ones through aggregation or pooling strategies [37]. Conversely, (natively) global methods can be usually applied to also predict specific instances.

2) Stage — Pre-model vs. Post-Hoc vs. Intrinsic: **Pre-model XAI techniques** encompass a range of strategies aimed at improving model transparency and interpretability before its deployment. For instance, Exploratory Data Analysis (EDA) involves delving into the dataset to uncover insights and potential biases, while feature engineering and selection help in selecting or transforming features for improved interpretability [38]. **Given the complexity and heterogeneity of the inputs used in the context of NTA, this phase plays an even more significant role in providing useful insights to network operators, as it ensures that the training data is clean and suitable for representing network traffic.** Techniques like bias and fairness analysis are essential to identify and address biases in the data and the model [39].

Post-hoc explainability refers to methods that analyze and interpret a pre-trained AI model [40]. These methods are applied externally to the model and usually do not rely on its internal workings. Post-hoc methods aim to provide insights into the decision-making process of the model by examining its inputs, outputs, or internal representations but without interfering with model training. These approaches represent the best choice when traffic analysis tools are already operating in real-world contexts, and there is a need to gain insight into their operations.

Intrinsic (or **by-design**) explainability, on the other hand, refers to methods that focus on building AI models or algorithms with built-in interpretability. These methods aim to

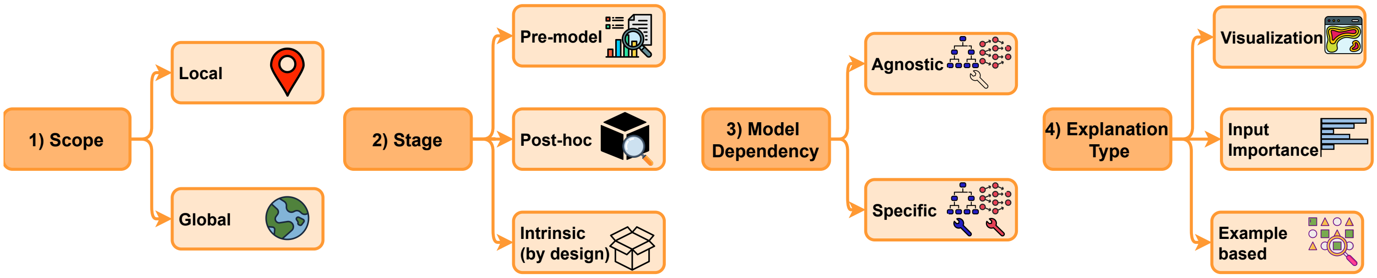


Figure 2. Different facets for XAI methods characterization.

create inherently transparent models and provide explanations alongside their predictions and often involve using simpler and more interpretable models or incorporating explicit rules and logic into the AI system's architecture. Explainable by-design models are the ideal choice when implementing new AI-driven traffic analysis systems, as they offer inherent transparency in their operations, rendering immediate interpretation of the system's decisions and fostering trust in its actions.

As an example of intrinsic explainability, attention mechanisms [41] can enhance neural network interpretability by dynamically assigning weights to distinct segments of input sequences, providing insights into which elements influence the decision-making process. Visualizing these attention weights reveals the model's focus, helping to understand its decision dynamics. This intrinsic transparency enables the designer to identify crucial elements that impact the model's outputs. Boolean Decision Rules via Column Generation (BRCG) [42] involves iteratively generating and adding simple Boolean decision rules to approximate the decision boundaries of a complex model. The optimization process selects the most informative rules in each iteration, and a pruning step helps improve generalization. The final set of Boolean decision rules provides a transparent and interpretable representation of the original model, offering explicit conditions for predictions. Another example is RuleFit [43], that is a XAI technique that begins by generating Decision Trees (DTs) and extracting rules from them. These rules, representing conditions on input features, are transformed into binary features. RuleFit combines these features with a linear model, such as logistic regression, allowing for the modeling of complex relationships.

3) Model Dependency — Specific vs Agnostic: **Model-specific** techniques are designed and tailored to a specific AI model or algorithm. These techniques take advantage of the specific characteristics, internal workings, and model structure to provide explanations and exploit the model's architecture, parameters, or intermediate representations. Considering, for example, a system used to classify network attacks based on a Convolutional Neural Network (CNN), employing an XAI method tailored to this type of network could be of paramount importance in situations where detailed explanations of internal model decisions are required from network administrators. These tools give insight into key aspects such as neurons' importance and layer activations, and enable a more accurate understanding of the model decisions. **Model-agnostic** techniques do not rely on the internal details or specifics of a particular AI model. These techniques are more generic and can be applied to any black-box model without requiring any knowledge about its internal workings. They typically treat the model as a black-box and use input-output mappings or other approximation methods to explain its behavior. Such methods are especially well-suited when different AI-based tools for traffic analysis are utilized and there is a need for a unified tool for explainability purposes (able to explain possible disagreements). This helps streamline the process, maintain consistency in interpreting results, and avoid the complexity and inefficiencies of managing multiple tools.

4) Explanations Types— Visualization vs. Input Importance vs. Example Based: As explainability is related to human understanding, the form in which the explanations are communicated becomes an important aspect. In this regard, the various state-of-the-art XAI techniques differ widely.

Some techniques provide explanations based on **Visualization**, relying on the intuitiveness of interpretation. Visual explanations often involve heatmaps, saliency maps [44], or other graphical representations to *highlight important features or regions in an input*. Among these, and despite not being an interpretability technique in the strict sense, t-distributed Stochastic Neighbor Embedding (t-SNE) [45] is often applied to the space of features extracted from DL models. Consequently, it becomes instrumental in *interpreting the representations* derived from the models and the results of classification tasks, offering insights into the separation capabilities of the analyzed model. **For instance, by visualizing the separation of normal traffic from various attack types, t-SNE offers valuable insights into the classification abilities of a network IDS.** Other visualization-based techniques are specific to CNNs and rely on the analysis of feature maps and neuron activations across different layers of the architecture (e.g., layers and feature maps visualization).

Many techniques focus on **Input Importance**: they aim at explaining model decisions based on the importance or contribution of individual inputs. For example, when interpreting a network IDS, input importance techniques can identify which traffic features (e.g., packet size or specific protocol flags) play the most critical role in classifying a flow as malicious or benign. Such insights are extremely valuable for network administrators in understanding the behaviors or patterns influencing the system's decisions. Among these techniques, *perturbation-based explanations* involve making changes to the input data and observing how the model's output changes [46]. By systematically perturbing the input, users can gain insights into the model's sensitivity and the importance of different features. A conceptually straightforward approach is provided by *occlusion analysis* [44], which iteratively assesses the models' performance by occluding (e.g., substituting with non informative data) portions of the inputs. The underlying rationale is that the more the model relies on a specific subset of inputs for predictions (i.e. the more the inputs are important), the more pronounced the variations in performance should be. Similarly, Permutation Feature Importance (PFI) [47] involves randomly shuffling the values of a specific feature and measuring the impact on the model's performance metrics. By comparing the model's original performance with the performance after permuting the feature, a decrease indicates that the feature is important for the model. SHapley Additive exPlanations (SHAP) [48] is based on cooperative game theory principles to fairly attribute the contribution of each feature. It considers all possible permutations of features, assessing how their inclusion or exclusion affects the model's output, and provides the average contribution of each feature. Other techniques are based on *approximating the model's behavior* in specific input regions using surrogate models that are more easily (or inherently) interpretable. Local Interpretable Model-agnostic Explanations (LIME) [49] works by generating perturbed samples and observing their corresponding predictions. It fits a simple, interpretable model, such as linear regression, to these perturbed samples, offering a transparent representation of the local decision boundary. Similarly, Local Explanation Method using Nonlinear Approximation (LEMNA) [50] produces a concise set of interpretable features to clarify the classification of an input data sample. *Gradient-based explanations* instead use information from the model's gradients to explain how model output would change if the input were modified [51]. Layer-wise Relevance Propagation (LRP) [52] assigns relevance scores to input features by systematically distributing relevance backward from the output layer to the input layer. These techniques are often model-agnostic as they rely on the input-output behavior of the models. Despite this, there are also examples of model-specific techniques such as Gradient-weighted Class Activation Mapping (Grad-CAM) [53], that are tailored for CNNs. Furthermore, certain techniques offer insights into the *significance of inputs by contrasting them with reference values* known as baselines (neutral feature values). Integrated Gradients (IG) [54] constructs a path from this baseline to the actual input, computing gradients at intermediate points along the way. Then, it integrates these gradients, attributing the model's output to each feature based on their contributions along the path. Deep Learning Important FeaTures (DeepLIFT) [55] computes the variance in the activation of each neuron between a given input and the baseline. These differences are then propagated backward through the network.

Example-based Explanations use specific instances to provide explanations. They often involve showing similar instances or explaining a decision by presenting a similar case that the model got right or wrong. When interpreting, for instance, a system used to recognize the application generating traffic flows, this type of explanation method can provide network administrators with highly intuitive insights. For example, if a flow is classified as originating from a video streaming app, the explanation could present similar flows the model has previously classified correctly, highlighting common characteristics such as packet size, volume, or traffic patterns shared among these flows. Counterfactual explanations [56] are valuable in scenarios where users want to understand why a model made a particular decision and what changes could lead to an alternative outcome. An example is represented by Contrastive Explanation Method (CEM) [57] that operates by contrasting the model's original prediction with an alternative outcome by generating a counterfactual instance, a modified version of the input that would lead to a different prediction while attempting to minimize changes. ProtoDash [58] serves as a method for identifying diverse prototypical examples (prototypes) that effectively encapsulate a comprehensive representation of the dataset. Explanations consist of the selected set of representative examples. Contrastive Autoencoder for Drifting detection and Explanation (CADE) [59] serves the purpose of identifying drifting samples that deviate from established classes and offers explanations to justify the detected drift. To elucidate the significance of the identified drift, a distance-based explanation method is employed. Contextual Outlier INterpretation (COIN) [60] describes the anomaly present in outliers identified by detectors. The interpretability of an outlier is established by considering three elements: the outlierness score, the attributes influencing the abnormality, and a contextual description of its neighborhoods.

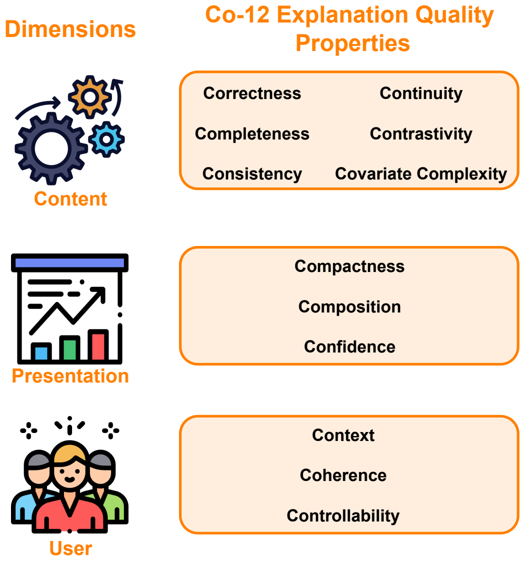


Figure 3. Properties for explanations quality assessment grouped by the three dimensions: Content, Presentation, and User.

All these techniques provide different perspectives: the choice of which XAI technique to use will depend on the specific use case, the type of model, and the goals of the explanation. To assess an XAI technique quality or compare different techniques, a number of metrics, summarized hereafter, can be adopted.

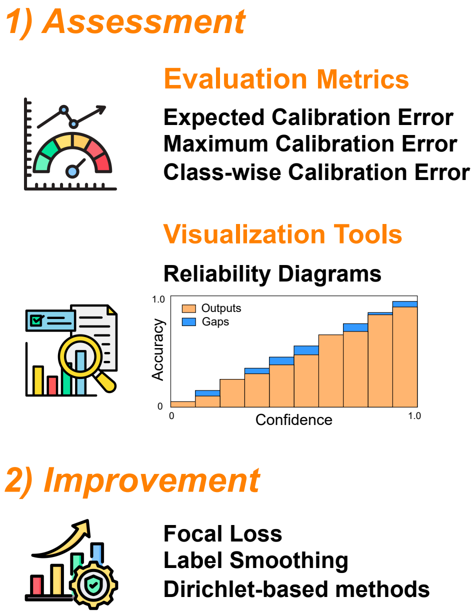


Figure 4. Core aspects of AI models calibration.

B. Metrics for Evaluating Quality of Explanations

While interpretability and explainability are often presented as subjectively validated binary properties, recent research has conceived it as a multifaceted and measurable concept [4]. Specifically, as depicted in Figure 3, *twelve properties* are identified for comprehensively assessing the quality of an explanation, grouped according to three dimensions: (i) *content* (e.g., how much explanation outcome is robust to slight variations), (ii) *presentation* (e.g., how much the explanation is simple and self-explaining), and (iii) *user* (e.g., how much the explanation is relevant to user needs and aligns to prior domain knowledge). For each of these properties, a set of verification scenarios/metrics can be defined. In [4] also emerges that, to date, the most considered explanation-quality properties are Coherence (user-related), Compactness (presentation-related), Completeness, and Correctness (both content-related).

Coherence evaluates the alignment of the explanation with domain knowledge (if a ground truth is available), or with other established XAI methods (used as gold standard). *Compactness* measures the size or sparsity of the produced explanation (the bigger the worse, as the user can be overwhelmed). (Output-)*Completeness* quantifies if the explanation holds enough information to explain the output. *Correctness* measures to what extent the explanation is faithful to the predictive model it explains. In our analysis of XAI applied to NTA it is later shown (Section V-A) that instead *Continuity* (content-related) is the most considered property in this domain. *Continuity* measures the generalizability of explanations (i.e. the effect on the explanation when slightly different inputs, not altering the model prediction, are considered).

For most of the twelve properties, different metrics and measurement scenarios can be considered: we refer to [4] for further details on the definitions and the associated quantitative techniques of each explanation-quality property.

C. Reliability of AI Models

When dealing with classification tasks and the probabilistic nature of ML and DL outputs, it becomes essential to inquire about the calibration of these probabilities. Indeed, in addition to explanation, the *confidence* of the classification must be reliable: attaching the probabilistic equivalent of "not much sure about this" or "little clues, it's mostly a random guess" to a classification response enables operators or cascading subsystems to deal with the response in a more informed way, as opposed to no confidence indication (amounting to "I'm positive about this" all the time), or worse, an unreliable (under/over-confident) indication. Therefore a key objective in AI algorithm design is to create a (black-box) model that can be trusted, ensuring its outputs meet specific reliability criteria. More specifically, *calibration* refers to the extent to which the model's confidence in its decisions accurately reflects its performance.

Various evaluation metrics that evaluate classification confidence have been introduced, and encompass different notions of calibration, such as multi-class, class-wise, and confidence-based [61]. Figure 4 introduces the two core aspects related to the calibration of AI-based models: assessment and improvement.

1) Assessment: Various **evaluation metrics** assess model calibration, such as Expected and Maximum Calibration Error, which only consider the confidence in the predicted class, and Class-wise Calibration Error, which instead considers the entire probabilistic output of the model.

Visualization tools such as reliability diagrams [62] and statistical tests are commonly used to verify and visualize the calibration properties of AI models. Reliability diagrams, for example, plot the accuracy as a function of confidence, providing a clear visual representation of this crucial aspect.

2) Improvement: Researchers are actively working on improving the calibration and trustworthiness of ML/DL models by utilizing advanced learning techniques to instill better calibration. Among these, Dirichlet-based methods [61], Focal Loss [63] and Label Smoothing [64] are prominent approaches. In particular, Focal Loss is originally designed to address class imbalance in classification tasks and has also been shown to improve calibration through its inherent entropy-based regularization properties. Label Smoothing acts as a form of loss regularization to enhance model generalization and prevent overconfident predictions. This method modifies the cross-entropy loss during training by minimizing it with respect to a smoothed one-hot representation of the ground truth.

Conversely, for prediction tasks reliability is only a recent concern, as classic predictors were based only on *point estimates*. Indeed, only recent AI literature has shifted to design of prediction approaches which have *uncertainty quantification* as a key objective, i.e. to describe predictive distributions over forecast variables for given inputs. Different techniques have been proposed for this more challenging case, ranging from ensemble methods, Bayesian methods, quantile regressors and, more recently, to conformal prediction [65]. The latter set of techniques has the advantage of not-making any restrictive assumption on the distribution of the forecast variable, i.e.

it is a class of *distribution-free* methods [66]. Accordingly, reliability improvement is not a separated concern, but rather these methods are conceived with the aim of being *reliable-by-design*. In the context of prediction, methods to assess the reliability of such techniques are mainly based on the calculation of concise metrics associated with *confidence intervals*, such as their mean length (viz. sharpness) and coverage w.r.t. the true one, or losses associated with *quantiles* (at different levels).

IV. XAI for NTA: Tasks

NTA is a critical process that involves collecting and examining network data to improve the performance and security of communication networks [34]. However, the nature of network traffic has evolved significantly in recent years, both in terms of volume and composition, introducing unprecedented challenges. Traditional approaches to NTA, such as Deep Packet Inspection, have become less effective due to the widespread adoption of cryptographic protocols that obscure packet content [169]. Additionally, the growing heterogeneity of network traffic further complicates the analysis and management of modern networks.

As discussed previously,in response to th, AI techniques, particularly those based on ML and DL [170], have emerged as powerful solutions. While ML models typically rely on features that are manually extracted from the data (such as packet statistics), DL approaches are capable of directly processing raw traffic data, making them particularly suited to handling the dynamic and constantly evolving nature of network traffic without requiring continuous manual feature engineering and model updates. For further background and details on ML and DL applied to NTA tasks, we refer the reader to dedicated overviews and surveys [2, 33, 70] (and references therein).

Despite their effectiveness, DL-based tools have a notable drawback: they often function as black boxes, making it difficult to understand the reasoning behind their predictions. This raises significant concerns about the interpretability of their outputs and the trustworthiness of the management policies they suggest. For network operators, it is essential to obtain not only accurate predictions from these models but also explanations of their behavior. Understanding the rationale behind a model's decisions is critical for operators to confidently implement the recommended network management policies.

To address this issue, XAI has been increasingly applied in the NTA domain. The efforts of the scientific community involved in the adoption of XAI approaches to NTA primarily focus on the NTA tasks discussed in the following. Table IV provides an overview, grouping works by the specific task faced.

A. Legit-Traffic Classification

Network Traffic Classification (TC) plays a crucial role in modern network management activities. It aims at associating the traffic flowing through a computer network with the (category of) applications that generated it [169].

By classifying network traffic, administrators can gain insights into the nature of the traffic and implement appropriate policies for network optimization. For instance, understanding the types of network traffic with the related patterns aids in optimizing network performance. This entails pinpointing bandwidth-intensive applications or traffic bottlenecks. In turn, this allows for allocating resources effectively, prioritizing critical applications, and ensuring a smooth user experience. Additionally, classifying network traffic enables the implementation of Quality of Service (QoS) policies, allowing different classes of traffic to be assigned varying levels of priority and resources, ensuring critical applications receive sufficient bandwidth and low latency. In cases compatible with *network neutrality* regulations, TC is also the basis for service/billing differentiation practices.

Formally, TC is a *multiclass classification* task and consists of assigning each Traffic Object (TO)—such as flows, biflows, service bursts, etc.—a label within the set $\{1, \cdots, L\}$, where each class corresponds to a specific traffic category. Most of the reviewed studies consider applications (also shortly referred to as *apps*) [67–80] or *traffic services* (traffic types, such as Email, Chat, FileTransfer, P2P, Streaming, VoIP) [72, 73, 76–78, 80–86], as possible classes. *Other* works differ in the considered classes and deal with website fingerprinting [88, 91], device classification [89], apps and user activities classification [74], or encapsulation identification [77, 85, 87]. The latter aims to distinguish whether a TO is collected through regular sessions or encapsulated within VPN sessions. Furthermore, the work [90] addresses multiple types of network analysis problems, including protocol identification, determining if a given Transport Layer Security (TLS) stream contains DNS-over-HTTPS, and identifying specific Remote Desktop Protocol and Secure Shell authentication methods. It is worth noting that certain studies tackle multiple classification tasks, therefore, they are reported in Table IV multiple times. These studies employ either multitask approaches (e.g., [77]) or separate single-task methods.

For instance, backing TC with XAI methods allows one to understand which protocol field or payload byte is responsible for selecting Facebook rather than YouTube as a classification verdict or whether the data-driven traffic classifier is confident enough in delivering its decisions.

B. Intrusion Detection

Intrusion Detection (ID) is a broad concept that can be applied to various data types to identify potentially malicious or anomalous activities. For example, ID systems may focus on system calls in software-based security contexts, as discussed in [171], to detect suspicious behaviors. However, in the context of network traffic, the primary goal of ID is to differentiate between legit network usage and anomalous (malicious) activities (i.e. *binary classification*) that may pose risks to the security, the performance, or the stability of the network [172]. Anomalies can be either specific activities known to be malicious or deviations from expected (normal) behavior which indicate potential security threats. Accordingly, there are different approaches to performing ID. One key distinction

Table IV
WORKS DEALING WITH XAI FOR NTA TASKS (BREAK-DOWN BY TASK, IN CHRONOLOGICAL ORDER WITHIN THE TASK).

NTA Task		Year	Papers
Legit-Traffic Classification	Apps	2019	[67]
		2020	[68], [69], [70]
		2021	[71]
		2022	[72], [73], [74]
		2023	[75], [76], [77], [78], [79], [80]
	Services	2019	[81]
		2021	[82]
		2022	[72], [73]
		2023	[83], [77], [76], [78], [80], [84]
		2024	[85], [86]
	Others	2022	[87], [88], [89], [74], [90]
		2023	[77], [91]
		2024	[85]
Intrusion Detection	Misuse Detection	2018	[92], [93]
		2020	[94], [95]
		2021	[96], [97], [98], [99], [100], [101], [102], [38], [103]
		2022	[104], [105], [106], [107], [108], [109], [110], [111], [112], [113], [114]
		2023	[115], [116], [117], [118], [119], [120], [121], [122]
		2024	[123], [124]
	Anomaly Detection	2019	[125], [126]
		2020	[127]
		2022	[128], [129], [112]
		2023	[130], [118]
	Others	2021	[131]
		2022	[132]
		2023	[133], [134]
Attack Classification	IoT Focused	2019	[135]
		2022	[104], [136], [137], [138], [139]
		2023	[116], [140], [141], [142], [143]
		2024	[123], [144]
	Non-IoT Focused	2016	[145]
		2018	[146]
		2020	[95], [147], [148], [149]
		2022	[87], [113], [150], [151], [152], [153], [154], [155], [156], [157]
		2023	[75], [115], [158], [159], [160]
		2024	[161]
Traffic Prediction		2021	[162], [163]
		2023	[164]
Other Tasks		2019	[165], [166]
		2020	[167], [168]

is between Misuse Detection (MD) and Anomaly Detection (AD) approaches.

Misuse Detection: In the MD task, both malicious and benign traffic data are available during model training. The task is typically formulated as a *supervised binary* (two-class) problem, where each traffic object is assigned a label from the set $\{\text{benign}, \text{malicious}\}$. The model learns to classify traffic instances based on labeled examples and seeks to generalize this classification to unseen data [92–102, 104–113, 115–118, 120–124]. Such approaches require labeled datasets that accurately represent both normal and anomalous network behavior.

Anomaly Detection: In the AD task, models are trained using only benign traffic data. The model learns the patterns and statistical properties of normal network behavior during training. Anomalies are then detected as deviations from the learned benign behavior. An AD approach does not require labeled anomalies during training (i.e. *unsupervised* learning) and is useful when labeled malicious data is scarce or when the characteristics of anomalous behavior are not well-defined [112, 118, 125–130].

Other Approaches: Some works do not strictly fall into the two subcategories discussed as they apply particular ad hoc methodologies to solve the ID task. Dias et al. [131] introduce a rule-based approach that integrates pre-existing rules created by experts with dynamically evolving knowledge generated by a DT algorithm, adapting to new insights gleaned from network activity. Jeong et al. [133] introduces a novel web-based visualization system that combines a range of visualization techniques to depict network traffic data. This system includes interactive features that empower users to engage in an interactive visual analysis of the data representation. The method presented in [132] utilizes self-supervised learning to train the model on unlabeled data for the purpose of identifying anomalies, determining the affected devices. Differently, Minh et al. [134] introduce a method that integrates various unsupervised models and combines them using a stacking strategy. From Table IV and the preceding discussion, it is evident that the majority of papers addressing ID tasks are concerned with MD.

For instance, applying XAI to ID allows operators to rank input data based on their actual contribution to the detection process. Also XAI enables them to pinpoint the portion of the data-driven model responsible for detecting specific anomalies (similarly to a rule-base detector).

C. Attack Classification

Attack Classification (AC) is concerned with identifying and categorizing specific malicious activities or cyberattacks within network traffic. The primary objective is to recognize and classify malicious activities or security threats, such as malware infections, denial-of-service attacks, intrusion attempts, phishing attacks, or any other unauthorized or harmful actions that can compromise the integrity, availability, or confidentiality of network resources. This task allows organizations to swiftly detect and respond to cyber threats, tailor their security measures to counteract known attack vectors, and improve overall cybersecurity by understanding and categorizing the nature of incoming threats targeting IoT devices (i.e. *IoT Focused*) [104, 116, 123, 135–144] or more general devices (i.e. *non-IoT Focused*) [75, 87, 95, 113, 115, 145–161].

This task results in a finer level of granularity in comparison to MD, as its objective is not only to differentiate between malicious and benign traffic but also to identify the specific class of attack responsible for generating the TO. More formally, AC is a *multiclass classification* task and consists of assigning each TO a label within the set $\{1, \cdots, L\}$, where each class corresponds to benign traffic or a specific attack. It is worth noting that the benign class may not be there and in these cases, the task aims to distinguish the various attacks from each other. Also, this task is similar to (legit) TC but it is characterized by distinct objectives and unique challenges associated with capturing and labeling network traffic [70].

Examining Table IV reveals a rising pattern in the number of works focused on AC, with a particularly notable surge in the last two years. XAI applied to AC helps gain insights into the specific features and patterns the model relies on to identify an attack, shedding light on the distinctive markers that set it apart from other attacks and the normal network behavior.

D. (Fine-grained) Traffic Prediction

Traffic Prediction (TP) refers to the practice of using historical data to forecast the patterns and trends of traffic flowing within computer networks. It involves analyzing the characteristics (e.g., volumes, rates) of network traffic to anticipate how it will evolve over time [173]. This prediction can encompass various aspects and can be performed at different granularity levels that reflect how network packets are aggregated, namely which is the TO constituting the elementary unit for the prediction task. Accurate TP enables network administrators to allocate resources such as bandwidth and processing power effectively. This prevents congestion, ensures optimal performance, and minimizes wastage of resources. Indeed, by understanding traffic growth patterns, organizations can scale their infrastructure to accommodate increasing demands without suffering from unexpected network saturation. Moreover, predicting network traffic aids in maintaining consistent QoS levels and ensuring that end-users receive reliable services, enhancing their overall experience.

Considering a generic TO, the TP task can be defined as the forecasting of a set of P traffic parameters (e.g., volume, bandwidth) based on a sequence of historical values assumed by these parameters. Such parameters are modeled as a multivariate time series characterized by a time index. Given observations of the parameters up to the n^{th} time index, TP aims to predict the traffic parameters associated with the $(n+1)^{th}$ time index. These predictions rely on past observations of the same traffic parameters, which act as the memory that the model must retain for making accurate predictions.

The TP task can be conducted at various granularities. Although most works typically focus on *aggregated metrics* (e.g., total volume and average rate, spanning over extended time intervals), recent years have seen a surge in the interest toward the challenging case of *fine-grained TP*. Nonetheless, the capitalization of DL for fine-grained TP is still a challenge, and performance gaps need to be overcome [163].

For the mentioned reasons, while some recent works deal with XAI on aggregated traffic prediction [174], the focus of this survey will be on investigating the adoption of XAI methods for TP performed at *packet-level* and *per-biflow* [162–164], i.e. given a biflow up to its n^{th} packet, TP aims to predict P traffic parameters associated with the $(n+1)^{th}$ packet. In this context, XAI methods can provide for instance detailed information about which element in a multivariate time series mostly contributes to correct predictions or about how many packets of a (bi)flow are helpful to obtain accurate inferences.

E. Other Tasks

Other studies do not fall within the set of tasks considered. In fact, they address distinct tasks, albeit related to the adoption of XAI to support the analysis of network traffic. Morichetta et al. [165] and Dethise et al. [166] focus on predicting video quality, while Terra et al. [167] address the prediction of SLA violations. Methods proposed by Meng et al. [168] were employed to interpret tasks related to video streaming, flow scheduling, and routing based on Software Defined Networking (SDN). We have reported these work in Table IV separately.

V. XAI for NTA: Goals

By definition, XAI aims at model explainability. This can serve a dual purpose: (i) it helps interpret the decisions of complex models, with the final goal of a better understanding for humans; (ii) it can further guide model improvement by revealing insights into areas where the model may need (architectural) adjustments or further (i.e. refined, targeted) training. In a practical application, the insights gained from XAI can inform a feedback loop where models are *iteratively* refined for better interpretability and performance.

Following this line, herein we first analyze the works that leverage XAI techniques to interpret the functioning of models built for NTA (Section V-A). Then, we discuss the works that utilize such insights obtained through XAI to refine the models, thus pursuing improvements along various directions (Section V-B). In doing so, we also summarize the *metrics* through which the benefits of applying XAI for both purposes are quantified.

A. XAI for Interpreting NTA Models

The literature analysis highlights how existing studies vary in the level of detail for the analyses conducted. In fact, many studies represent a **preliminary effort** in leveraging XAI and limit explainability investigations mainly to characterize the significance of inputs [67, 81, 83, 84, 91–93, 99, 100, 103–106, 113, 117, 119–122, 128, 130, 136–138, 140, 142, 145, 151, 155, 158–160, 162, 163, 175]. Despite the adoption of different methodologies, ranging from traditional to complex and innovative approaches, such studies mostly provide insights into the importance of inputs for decision-making models, highlighting how specific portions of the inputs (e.g., whole packets, packet fields, payload portions, handcrafted features) lead the model toward either correct or wrong outcomes.

Other studies offer **more detailed analyses**, focusing on aspects such as the errors made by the models [69] or investigating input importance for different models [146] or different sets of features and datasets together [109] or examining multiple perspectives on models [68, 80, 88].

Some research aims to introduce **new frameworks or methodologies to assist users in interpreting their models** [38, 87, 133, 147, 165, 168] or **novel XAI techniques**, as done by Zhang et al. [157], which propose a novel XAI technique based on Shapley Values to accelerate global explanations for IDSs.

A significant number of studies propose **new tools or instruments that are directly interpretable** by end-users [75, 76, 79, 94, 97, 126, 132, 134, 149, 150, 153, 176, 177]. Other research delves into insights obtained by investigating the explanations provided by various techniques, often **comparing different techniques** qualitatively [82, 90, 95, 98, 102, 108, 127, 167]. This comparison is frequently made with more traditional methods for feature selection [135, 148, 152].

Other works leverage XAI aiming at **interpreting specific facets of training, models, or data**. For example, several papers present analyses involving multiple datasets in a cross-evaluation context. Yilmaz and Bardak [110] aim to understand the difference between multiple datasets from the perspective of feature importance. Similarly, Layeghy and Portmann [112, 118] investigate the differences in feature importance for various combinations of training and test sets coming from different datasets, and compare them to the case where both training and test originate from the same dataset. Nascita et al. [178] employ various explainability methodologies to understand the differences in the behavior of traffic classifiers based on the training methodology. These classifiers are trained either traditionally (having access to all class data during training) or incrementally (using class incremental learning techniques). In a similar context but with a different purpose, Jorgensen et al. [78] evaluate their models with out-of-distribution detection analysis to understand when to retrain the model from scratch. Nascita et al. [139] utilize XAI to evaluate the impact of specific network-layer fields and the potential biases they introduce on the performance.

Concerning **reliability** in TC tasks, Li et al. [72] propose a trustworthy TC model that provides both classification predictions and confidence scores, aiding in the effective differentiation between correct and incorrect predictions, while Guarino et al. [74] examine model calibration to characterize approaches from this perspective, in addition to traditional performance-related aspects. As for reliability in TP tasks, in [74] the authors face a fine-grained TP task and investigate the reliability in the prediction of a discrete parameter (i.e., the packet direction) by framing it as a binary classification task. Conversely, direct reliability enforcement in fine-grained network prediction tasks seems unexplored to the best of authors' knowledge.

Only a few works in NTA defines and/or utilizes **metrics for evaluating explanations** [75, 96, 115, 143, 150]. Table V

Table V
SUMMARY OF EXPLANATION-QUALITY METRICS AS IN THE NTA LITERATURE. THE METRICS ARE ORGANIZED IN ALPHABETICAL ORDER (↓).

Metric in NTA ↓	**Descriptions (as in NTA literature), references, and formulas**	**Category as in [4]**	**Notes on Applicability to XAI Methods**	**Application to NTA Tasks**			
				TC	**AC**	**ID**	**TP**
Accuracy (✚)	Quantifies the *accuracy* of global explanations by using its top-k most significant features [115]. $= \mathrm{acc}(\mathcal{M}) - \mathrm{acc}(\mathcal{M}_{\mathrm{top-k}})$, where $\mathcal{M}_{\mathrm{top-k}}$ is the model using only top-k features highlighted by a *global* explanation (i.e. $\bar{\mathcal{A}}_{\mathrm{top-k}}(\mathcal{M}, \mathrm{xm})$)	*Correctness*	+G*	⤢	✓	✓	⤢
Compactness (━)	Quantifies (via *inertia*) the cohesiveness of explanations within the same class [150]. $= (1/L) \sum_{\ell=1}^{L} (1 / \lvert\mathcal{D}_\ell\rvert) \sum_{\boldsymbol{x} \in \mathcal{D}_\ell} \lVert \boldsymbol{a}(\mathcal{M}, \boldsymbol{x}, \mathrm{xm}) - \bar{\boldsymbol{a}}_\ell(\mathcal{M}, \mathrm{xm}) \rVert^2$, where $\bar{\boldsymbol{a}}_\ell(\mathcal{M}, \mathrm{xm})$ denotes the centroid explanation of all the samples belonging to class ℓ, i.e. $\mathcal{D}_\ell$. Metric can be averaged on either training or test sets.	*Continuity*	+EB	⤢	✓	✓	✗
Consistency (🖼)	*Visually assesses* how consistent the explanations are for similar instances across different models, regardless of the model used [115]. $\boldsymbol{a}(\mathcal{M}_j, \boldsymbol{x}, \mathrm{xm})$ vs. $\boldsymbol{a}(\mathcal{M}_j, \tilde{\boldsymbol{x}}, \mathrm{xm})$ (or $\mathcal{A}_{\mathrm{top-k}}(\mathcal{M}_j, \boldsymbol{x}, \mathrm{xm})$ vs. $\mathcal{A}_{\mathrm{top-k}}(\mathcal{M}_j, \tilde{\boldsymbol{x}}, \mathrm{xm})$) for different models $\mathcal{M}_j$, $j = 1, \ldots, M$.	*Consistency*	+EB	⤢	✓	✓	⤢
Effectiveness (✚)	Quantifies (as a *binary value*) whether the explanation results are important to the decision-making process and thus change the classification results [75]. $= 1$ (resp. $= 0$) if $\boldsymbol{x}$ is mutated in $\boldsymbol{x}_m$ by modifying all the features contained in $\mathcal{A}_{\mathrm{top-k}}(\mathcal{M}, \boldsymbol{x}, \mathrm{xm})$ and $\mathcal{M}(\boldsymbol{x}_m) \neq \mathcal{M}(\boldsymbol{x})$ (resp. $\mathcal{M}(\boldsymbol{x}_m) = \mathcal{M}(\boldsymbol{x})$). The metric is averaged over the entire dataset.	*Correctness*		⤢	✓	✓	✗
Efficiency (━)	Quantifies (via *runtime*) if interpretations are promptly available in high-speed online workflows [96]. $\mathrm{runtime} = \sum_{i \in \mathcal{D}^\circ} \Delta t_i$, where Δt_i is the time needed to generate the ith explanation and $\mathcal{D}^\circ$ can be the whole dataset ($\mathcal{D}$) or a subset of it.	*N/A*	+EB	⤢	✓	✓	⤢
Efficiency (━)	Quantifies (via *latency*) if interpretations do not introduce significant delays in high-speed online workflows [143]. $\mathrm{latency} = \frac{1}{\lvert\mathcal{D}^\circ\rvert} \sum_{i \in \mathcal{D}^\circ} \Delta t_i$, where Δt_i is the time needed to generate the ith explanation and $\mathcal{D}^\circ$ can be the whole dataset ($\mathcal{D}$) or a subset of it.	*N/A*	+EB	⤢	✓	✓	⤢
Fidelity (✚)	Measures (via *Label Flipping Rate, LFR*) the proportion of anomalies that become normal when considering only a fraction of the input features, according to their importance [96]. LFR is the ratio of misclassified samples to those originally classified as correct when inputs corresponding to $\mathcal{A}_{\mathrm{top-k}}(\mathcal{M}, \boldsymbol{x}, \mathrm{xm})$ are nullified for each $\boldsymbol{x}$.	*Correctness*		⤢	⤢	✓	✗
Reliability (✚)	Quantifies the similarity (via *Dice-Sørensen coefficient*) between the explanations of different XAI techniques (xm_1 vs. xm_2) used on the same model and the same samples [143]. $= \mathrm{DS}(\mathcal{A}_{\mathrm{top-k}}(\mathcal{M}, \boldsymbol{x}, \mathrm{xm}_1), \mathcal{A}_{\mathrm{top-k}}(\mathcal{M}, \boldsymbol{x}, \mathrm{xm}_2))$. The metric is averaged over the entire dataset.	*Correctness*		⤢	⤢	✓	✗
Reliability (━)	Quantifies (via *Average Reliability*) the differences in feature importance scores between different XAI methods (xm_1 vs. xm_2) and indicates how consistent the model explanations are [143]. $= \lvert \boldsymbol{a}(\mathcal{M}, \boldsymbol{x}, \mathrm{xm}_1) - \boldsymbol{a}(\mathcal{M}, \boldsymbol{x}, \mathrm{xm}_2) \rvert_1 / N$. The metric is averaged over the entire dataset.	*Correctness*	+EB	⤢	✓	✓	⤢

Metric Interpretation: (✚): the higher the better; (━): the lower the better; (🖼): for visualization-based metrics an ordering criterion can not be defined.
Metric Formulas: $\boldsymbol{a}(\mathcal{M}, \boldsymbol{x}, \mathrm{xm})$ denotes the local explanation obtained from interpretability method xm on model $\mathcal{M}$ and sample $\boldsymbol{x}$ (size N), whereas $\mathcal{A}_{\mathrm{top-k}}(\mathcal{M}, \boldsymbol{x}, \mathrm{xm})$ denotes the corresponding set of top-k features in case the xm method falls within the Input Importance category. $\bar{\mathcal{A}}_{\mathrm{top-k}}(\mathcal{M}, \mathrm{xm})$ has the same meaning, but for a global explanation. $\mathrm{JS}(\mathcal{A}, \mathcal{B}) = \lvert\mathcal{A} \cap \mathcal{B}\rvert / \lvert\mathcal{A} \cup \mathcal{B}\rvert$ while $\mathrm{DS}(\mathcal{A}, \mathcal{B}) = 2\lvert\mathcal{A} \cap \mathcal{B}\rvert / (\lvert\mathcal{A}\rvert + \lvert\mathcal{B}\rvert)$, where $\mathcal{A}$ and $\mathcal{B}$ are two generic sets.
XAI methods: All metrics are applicable to all Local XAI methods and Global methods that allow one to extract explanations for single samples. A subset of them is applicable *also* to Global XAI methods that cannot allow to extract explanations on single instances (**+G***). All metrics are applicable to Input Importance-based XAI methods. A subset of them is applicable *also* on Example-based XAI methods (**+EB**). All metrics are applicable to the following variations of XAI methods: Model-Agnostic/Specific, Post-Hoc/Intrinsic. Pre-Model methods are not suitable for explanation-quality assessment.
Application to NTA Tasks: ✓: applied to / proposed for the NTA task; ⤢: adaptable to the NTA task possibly with minor modifications; ✗: not adaptable to the NTA task with minor modifications.

(Continued) Table V

SUMMARY OF EXPLANATION-QUALITY METRICS AS IN THE NTA LITERATURE. THE METRICS ARE ORGANIZED IN ALPHABETICAL ORDER (↓).

Metric in NTA ↓	Descriptions (as in NTA literature), references, and formulas	Category as in [4]	Notes on Applicability to XAI Methods	Application to NTA Tasks			
				TC	AC	ID	TP
Robustness (+)	Quantifies (via *Jaccard Similarity*) the robustness of explanations to noise [96]. JS($\mathcal{A}_{\text{top-k}}(\mathcal{M}, \boldsymbol{x}, \text{xm}), \mathcal{A}_{\text{top-k}}(\mathcal{M}, \boldsymbol{x} + \boldsymbol{n}, \text{xm})$), where $n_i \sim \mathcal{N}(0, \sigma^2)$. The metric is averaged over the entire dataset.	*Continuity*		⤢	⤢	✓	⤢
Robustness (+)	Measures (via *Dice-Sørensen coefficient*) how similar the explanation results are for similar instances [75]. $= \overline{\text{DS}}_1 - \overline{\text{DS}}_2$, where $\overline{\text{DS}}_1$ (resp. $\overline{\text{DS}}_2$) represents the average of DS($\mathcal{A}_{\text{top-k}}(\mathcal{M}, \boldsymbol{x}, \text{xm}), \mathcal{A}_{\text{top-k}}(\mathcal{M}, \tilde{\boldsymbol{x}}, \text{xm})$) over all the samples $\tilde{\boldsymbol{x}}$ in the dataset having the same prediction as (resp. different prediction from) $\mathcal{M}(\boldsymbol{x})$.	*Continuity*	+EB	⤢	✓	✓	✗
Robustness (−)	Measures at the per-class consistency of explanations between the training and testing datasets (via *Average Link Distance*) [150]. $= \lvert \boldsymbol{a}(\mathcal{M}, \boldsymbol{x}, \text{xm}_1) - \boldsymbol{a}(\mathcal{M}, \boldsymbol{x}, \text{xm}_2) \rvert_1 \,/\, N$. The metric is averaged over the entire dataset.	*Continuity*	+EB	⤢	✓	✓	⤢
Separability (+)	Quantifies (via *differential Accuracy*) whether explanations themselves can help differentiate between different classes effectively [150]. $= \text{acc}(\mathcal{M}^{\text{in}}_{\text{clus}}) - \text{acc}(\mathcal{M}^{\text{exp}}_{\text{clus}})$, where $\mathcal{M}^{\text{in}}_{\text{clus}}$ (resp. $\mathcal{M}^{\text{exp}}_{\text{clus}}$) represents the clustering-based classifier working on original training data (resp. explanations taken from them).	*Continuity*	+EB	⤢	⤢	✓	✗
Separability (+)	Quantifies (via *differential F1-score*) whether explanations themselves can help differentiate between different classes effectively [150]. $= \text{F1}(\mathcal{M}^{\text{in}}_{\text{clus}}) - \text{F1}(\mathcal{M}^{\text{exp}}_{\text{clus}})$, where $\mathcal{M}^{\text{in}}_{\text{clus}}$ (resp. $\mathcal{M}^{\text{exp}}_{\text{clus}}$) represents the clustering-based classifier working on original training data (resp. explanations taken from them).	*Continuity*	+EB	⤢	⤢	✓	✗
Stability (🖼)	*Visually assesses* how consistent the explanations are for similar instances within a single model [115]. $\boldsymbol{a}(\mathcal{M}, \boldsymbol{x}, \text{xm})$ vs. $\boldsymbol{a}(\mathcal{M}, \tilde{\boldsymbol{x}}, \text{xm})$ (or $\mathcal{A}_{\text{top-k}}(\mathcal{M}, \boldsymbol{x}, \text{xm})$ vs. $\mathcal{A}_{\text{top-k}}(\mathcal{M}, \tilde{\boldsymbol{x}}, \text{xm})$).	*Continuity*	+EB	⤢	✓	✓	⤢
Stability (+)	Quantifies (via *Jaccard Similarity*) the similarity of explanations provided for the same samples across multiple runs [96]. JS($\mathcal{A}^{(1)}_{\text{top-k}}(\mathcal{M}, \boldsymbol{x}, \text{xm}), \mathcal{A}^{(2)}_{\text{top-k}}(\mathcal{M}, \boldsymbol{x}, \text{xm})$), where superscript in $\mathcal{A}_{\text{top-k}}$ refers to two different runs of explanation with the same setting. The metric is averaged over the entire dataset.	*Continuity*		⤢	⤢	✓	⤢
Stability (+)	Reflects that the input importance should not be significantly affected by small changes to the model (via *Dice-Sørensen coefficient*), such as the number of epochs [75]. $= \lvert \boldsymbol{a}(\mathcal{M}, \boldsymbol{x}, \text{xm}_1) - \boldsymbol{a}(\mathcal{M}, \boldsymbol{x}, \text{xm}_2) \rvert_1 \,/\, N$. The metric is averaged over the entire dataset.	*Correctness*	+EB	⤢	✓	✓	⤢

Metric Interpretation: (+): the higher the better; (−): the lower the better; (🖼): for visualization-based metrics an ordering criterion can not be defined.

Metric Formulas: $\boldsymbol{a}(\mathcal{M}, \boldsymbol{x}, \text{xm})$ denotes the local explanation obtained from interpretability method xm on model $\mathcal{M}$ and sample $\boldsymbol{x}$ (size N), whereas $\mathcal{A}_{\text{top-k}}(\mathcal{M}, \boldsymbol{x}, \text{xm})$ denotes the corresponding set of top-k features in case the xm method falls within the Input Importance category. $\bar{\mathcal{A}}_{\text{top-k}}(\mathcal{M}, \text{xm})$ has the same meaning, but for a global explanation. $\text{JS}(\mathcal{A}, \mathcal{B}) = |\mathcal{A} \cap \mathcal{B}| \,/\, |\mathcal{A} \cup \mathcal{B}|$ while $\text{DS}(\mathcal{A}, \mathcal{B}) = 2|\mathcal{A} \cap \mathcal{B}| / (|\mathcal{A}| + |\mathcal{B}|)$, where $\mathcal{A}$ and $\mathcal{B}$ are two generic sets.

XAI methods: All metrics are applicable to all Local XAI methods and Global methods that allow one to extract explanations for single samples. A subset of them is applicable *also* to Global XAI methods that cannot allow to extract explanations on single instances (**+G***). All metrics are applicable to Input Importance-based XAI methods. A subset of them is applicable *also* on Example-based XAI methods (**+EB**). All metrics are applicable to the following variations of XAI methods: Model-Agnostic/Specific, Post-Hoc/Intrinsic. Pre-Model methods are not suitable for explanation-quality assessment.

Application to NTA Tasks: ✓: applied to / proposed for the NTA task; ⤢: adaptable to the NTA task possibly with minor modifications; ✗: not adaptable to the NTA task with minor modifications.

lists the metrics adopted (and possibly introduced) in such works, along with their definitions and formulas we have extracted and harmonized based on the scattered descriptions provided in the reported references. Also, for the sake of a comprehensive overview, for each metric, we report: (i) how it measures explanation quality (i.e. the higher the better, or vice versa); (ii) the "translation" into the systematic categories proposed in [4]; (iii) the specific characteristics of XAI methods to which it applies, based on the dimensions presented in Sec. III-A; (iv) the NTA tasks it has been used for or could be adapted to with minimal adjustments.

Detailing, from the inspection of the table, one can notice how different studies consider diverse (sets of) metrics, the most common being *Robustness* and *Stability*. More important, the lack of consensus and clarity in metric definitions can be noted. In fact, different papers utilize identical names while referring to different metrics, or even distinct explanation aspects altogether (as in the case of *Stability*). This lack of agreement complicates the task of comparing and evaluating various explanation methods. Broadly speaking, the evaluation of XAI methods based on the quality of their explanations is of paramount importance to the scientific community. Hence, recent efforts have been dedicated to proposing or organizing existing metrics, laying the groundwork for evaluation frameworks that consider this fundamental aspect when designing a system with XAI components [4]. In line with these efforts, in Table V we have also provided a "translation" of the adopted terms and the corresponding ones used in the systematic organization presented in [4]. In the papers we analyzed in this survey, we identified 17 definitions for explanation-quality metrics being considered in NTA. All metrics can be applied to local XAI methods, and global methods that allow for the extraction of explanations for individual samples. Only one (*Accuracy*) is also applicable to global XAI methods that do not provide explanations for single instances (**+G***). Moreover, all the metrics can be used with input importance-based XAI methods, and a subset of them is also applicable to example-based XAI methods (**+EB**). Finally, all metrics apply to both model-agnostic and model-specific methods, and post-hoc and intrinsic methods. Pre-model methods are unsuitable for explanation-quality assessment, as they pertain to a stage where models have not yet been implemented, and therefore no explanations of their behavior are available. In general, despite adopting different techniques, a significant portion of these metrics pertain to the *Continuity* property (7 out of 17 definitions). The second most considered property is *Correctness* (4 out of 17 definitions), and both belong to the *content* viewpoint. This insight suggests that in NTA many explanation-quality properties are currently neglected, and future works should assess also other properties, at least covering the three viewpoints identified in [4], namely content, presentation, and user. To foster comparability of different models and explanations, we also suggest that a reduced set of shared metrics definitions, possibly tailored to the NTA domain, should be also adopted.

B. XAI to Improve NTA tools

The insights gained from XAI techniques can be employed for improving the NTA data-driven models from different perspectives. Indeed, the explanations that XAI techniques provide allow researchers to identify areas where AI models underperform or even make erroneous decisions. This enables iterative refinement, leading to more accurate, reliable, and robust NTA tools.

One potential strategy to improve performance is to **refine the inputs**. XAI techniques can identify the most important inputs as well as those with null or negative impact on the predictions. Thus, it is possible to attain performance enhancement by using only the inputs marked as essential and discarding those that could confuse the model leading to wrong outcomes. Even though several studies do not specifically aim to optimize models from the perspective of inputs, they assess the performance achieved when only subsets of the original inputs are utilized. This type of analysis holds dual value: it serves not only to optimize approaches and make them more efficient but also to validate that the features highlighted by XAI techniques are the most important for the model. For instance, Roshan and Zafar [129] optimize an AutoEncoder (AE) by using the most crucial 40 features (out of 78) based on SHAP and obtain an *optimized* model with higher area under the curve and accuracy w.r.t. the initial model. Garcia et al. [73] apply the same approach to traffic-type and app classification tasks. The initial model considers the first 1500 bytes of each packet as input. In both cases, the interpretability results indicate that information in preprocessed packet headers is the most important, while payloads are not relevant. To validate this result, the authors train the model using only the headers as input, and the results show small losses or even performance improvements. By doing so, they reduce input dimension and speed up the model via XAI. Keshk et al. [116] evaluate the performance achieved when utilizing only the top-20 features highlighted by SHAP or PFI, or by combining insights from both techniques. Notably, they also investigate how much training and detection times are reduced. He et al. [82] and Dethise et al. [166] demonstrate the effectiveness of focusing on a subset of features selected through XAI techniques, obtaining similar results as using the original inputs in terms of accuracy and quality of experience, respectively. Other studies employ **explanations as input to design innovative traffic analysis approaches** that are more accurate and robust. Caforio et al. [111] capitalize on the explanations to improve the system's accuracy and robustness by using them for classification through a Grad-CAM-based nearest-neighbor classification of network anomalies. Barnard et al. [107] propose a two-stage pipeline for the MD task. In the first stage, they employ an XGBoost model and utilize SHAP to generate explanations. In the second stage, these explanations are fed as input to an AE, which aims to learn a latent representation of typical behavior during training. Instances are then classified based on the reconstruction error of the AE assuming deviations from this baseline indicate zero-day attacks. Essentially, the final classification stage utilizes the explanations obtained in the initial stage as input. A similar approach is introduced also by Fujita et al. [114]. Malik et al. [154] propose a XAI-based fine-tuning procedure to improve accuracy and robustness to adversarial samples in cyber-threat detection. It consists of a two-step fine-tuning. After training on original and adversarial samples, the authors first fine-tune the model with the information produced by SHAP and then further fine-tune it using the adversarial training set. In this manner, they utilize explanations to help the model focus on the input features it identifies as the most relevant by designing a sort of dynamic feature selection on each sample. Dias et al. [131] pursue a distinct objective and present a methodology to achieve heightened and enduring security. They leverage a DT to uncover novel insights from observed network activity and expand the rules system, by combining rules derived from interpretability analysis with those initially provided by domain experts. Similarly, Li et al. [125] select the most important features and employ them to assist in generating network access control policies.

Another group of studies employs XAI to **enhance the feasibility** of approaches for analyzing Internet traffic. Actually, all studies that reduce the inputs of a model, even though not explicitly stated, enhance its feasibility. In general, a DL model with fewer inputs is less complex both structurally and in terms of training times. Moreover, in specific cases where the reduction is done judiciously and the nature of the input allows for it, this reduction also leads to a decrease in the "time-to-insight" (e.g., the time needed to gather the necessary number of packets or bytes for the model to produce the output). An example of this is pursued by Nascita et al. [77], leveraging the outcomes of explainability analyses to pinpoint the subset of inputs most pivotal for predictions. Subsequently, they exclusively employ this subset, reducing training times by $\approx 60\%$ and achieving a shorter time-to-insight. This efficacy arises from the system's need for a more restricted input set (i.e. a significantly lower number of packets) to deliver classification output. An alternative strategy for making a Deep Neural Network (DNN) deployable on resource-constrained systems involves *extracting rules using XAI techniques*. Yan et al. [101] implement this approach to devise an online system based on Field Programmable Gate Array (FPGA) tailored for high-speed network environments. They illustrate that integrating rules into an FPGA is not only straightforward but also more cost-effective than embedding DNNs. Wang et al. [89] follow a similar approach: after analyzing the interpretation outcomes for their model, they select specific key bytes and their positions within the packet to create pattern strings for TC. This approach not only maintains high accuracy levels but also significantly enhances the model's feasibility, making it a more viable and efficient solution for real-world applications since matching the pattern strings with flows requires considerably less computational effort.

From a **reliability standpoint**, some works go beyond calibration assessment [70, 74] and implement strategies to enhance models from this perspective. The goal is to obtain models where confidence better aligns with actual performance. In detail, Nascita et al. [71, 77] enhance model calibration by employing alternative loss functions (distinct from the classic cross-entropy) to mitigate the overconfidence observed in DL models. Notably, they improve the generalization capability of TC models using Focal Loss (FL) and Label Smoothing (LS) [71] and assess and improve the impact on calibration (and performance) of compression techniques such as knowledge distillation, pruning, and quantization [77].

From the analysis of literature on the usage of XAI to improve performance, ultimately it emerges that, while the insights gained from explainability analyses can provide valuable guidance on how to improve approaches from various perspectives, they do not (at their current stage of maturity) directly and automatically contribute to enhancing traffic detection or prediction accuracy. This remains an open challenge, discussed in Sec. IX-A.

VI. XAI for NTA: Practical Use Cases

The present section delves into the examination of practical use cases of XAI approaches employed in works addressing NTA tasks with a focus on *interpretability* and *reliability*.

A. NTA Works on Interpretability

Hereinafter, we discuss several works addressing *interpretability* in Internet NTA by following the characterization introduced in Section III-A. We underline that the different aspects covered *are not orthogonal*, hence we categorize the works based on their prominent aspects related to the NTA use case they take into account.

1) Categorization by Scope: Considering the scope of the explanations, most of the works consider XAI approaches providing **local** explanations for individual predictions. Nevertheless, a handful of recent works aim to achieve a **global** explanation of the overall behavior of considered models. In the domain of network security and privacy, Jacobs et al. [87] return global explanations via DT-distillation of black-box ML models, while Ables et al. [106] propose an explainable IDS based on Self Organizing Maps (SOMs) that can provide both global and local explanatory visualizations. Similarly, the *SPIP* framework proposed by Keshk et al. [116] and the IoT-related IDSs designed by Abou El Houda et al. [105, 108] can also produce both types of explanations by leveraging multiple XAI tools. Focusing on SHAP explanations, Šarčević et al. [152] discuss their different aggregations and overall visualizations capable of providing a global viewpoint in addition to local ones. Furthermore, Nascita et al. [71, 77, 178] pool the outcomes of local approaches (i.e. SHAP and IG) to obtain global explanations associated with different granularities (e.g., overall dataset, per-protocol, per-application). A slightly more sophisticated approach is pursued by Zhang et al. [157], who propose a new methodology for sub-sampling and pooling Shapley values to design a novel Shapley-based lightweight global explainer.

2) Categorization by Stage: XAI approaches can be applied at different stages of the model (to be interpreted) lifetime. The application stage not only impacts the working principle of the interpretability approach but can also enforce some constraints on the underlying model.

Pre-model. Interestingly, while most of the works apply *pre-processing operations* to refine data before exploiting the model for NTA, they do not claim to perform such operations to explicitly improve the transparency and interoperability of the model. A notable exception is the work by Liu et al. [38], which specifically devises a pre-modeling explainability module aiming at improving the quality of network traffic data (via data cleaning and redundant input removal) to aid the module in providing the actual explanations of model internals. Similarly, Islam and Eberle [113] propose a domain knowledge-aided system to improve the explainability of an IDS. Their method infuses the "confidentiality, integrity, and availability" principles within the model by mapping the selected features with the most related principle(s) and thus aiding the interpretability of the IDS. Finally, although not explicitly described as a pre-model solution, Piet et al. [90] present *GGFAST*, a framework that automates feature engineering to develop fast and interpretable NTA tools for different purposes, such as identifying protocols, finding DNS-over-HTTPS in TLS flows, and discovering SSH authentication method. GGFAST looks for snippets (i.e. characteristic patterns of message lengths) that give a way to characterize each class by underlying protocol or message idioms, thus revealing the features that make each class unique and potentially providing explainable decisions.

Post-hoc. Most of the considered works achieve explainability through *post-hoc* techniques, intending to identify the parts of the input that mostly influence the model's decision (i.e. input-importance techniques according to the categorization by "explanation type" defined in Section III-A). The primary reason for resorting to post-hoc methods is their applicability to pre-trained models, which do not require any modifications to the latter. Furthermore, since they are often model-agnostic, they can be applied across various model types.

Delving into the specifics, undoubtedly the most frequently employed technique is SHAP (in all its flavors), utilized in at least one analysis within many studies [69, 71, 77, 82–85, 98, 99, 103–105, 107–109, 112, 115–124, 128–130, 136–138, 141–144, 148, 151, 151, 152, 154–156, 158–160, 164, 167, 178]. Although most studies do not specify the exact SHAP implementation they used, some provide this detail by explicitly mentioning the use of versions specifically designed for the models being interpreted, like DeepSHAP [69, 71, 77, 84, 121, 144, 164, 178] and TreeSHAP [107, 152, 167], or the more general KernelSHAP [128, 128, 129].

The LIME technique is the next most frequently employed interpretability tool, and it is also used in a significant number of NTA-related works [82, 85, 89, 96, 98, 105, 114, 115, 117, 120, 121, 123, 124, 135, 136, 140, 141, 143, 160, 165–167]. These two post-hoc techniques are often used also together within the same work to analyze the same model from different perspectives or to conduct multiple separate interpretability analyses [82, 85, 98, 105, 115, 117, 120, 121, 123, 124, 136, 141, 143, 147, 160, 167].

Other post-hoc techniques introduced in Section III-A are used in fewer works facing Internet NTA. In more detail, LRP is employed by Amarasinghe et al. [92] and Amarasinghe and Manic [146]. The work by Nascita et al. [77] is the only one employing IG to compare their interpretability outcomes with those provided by SHAP, while Garcia et al. [73] use saliency maps to trace the most important inputs for per-packet TC at different granularity (i.e. traffic type and specific application). A similar approach called *GEE (Gradient-based Explainable Variational autoEncoder)* is proposed by Nguyen et al. [126] to explain the anomalies detected by a variational AE via a gradient-based fingerprinting technique. Grad-CAM is used by Caforio et al. [111] to interpret the decisions of the proposed *GRACE (GRad-CAM-enhAnced Convolution neural nEtwork)* IDS which performs a K-means clustering on localization maps explaining a CNN used for classification and by [86] for interpreting CNNs encrypted network packet classification results. Li et al. [125] propose an IDS for AD and generation of SDN flow rules, which operates based on the explanations provided by LEMNA. Lastly, Abou El Houda et al. [105, 108] design a XAI-powered framework that gives explanations to DL-based decisions for IoT-related IDSs exploiting SHAP, RuleFit, and LIME techniques. Similarly, Islam et al. [124] employ LIME, SHAP, ELI5, and ProtoDash, while Arikkat et al. [85] use SHAP, LIME, Permutation Importance, and Counterfactual Explanations.

Another line of research in Internet NTA employs post-hoc techniques that capitalize *both* the pre-trained model and the traffic dataset used for training it. These techniques are however usually tied to the selection of a specific surrogate (explanation) model. For instance, some works have taken a first step towards interpreting DL-based predictors of network traffic through Markovian Distillation [162, 163], aiming at emphasizing the variations in their behaviors. To this end, the authors distill small-order Markov Chains from the DL models and scrutinize the disparities in their predictive behaviors. A different post-hoc method that extracts tree-based rules to explain the relationship between input, hidden, and output layers of a pre-trained DNN is proposed by Yan et al. [101]. The goal is to shed light on the decision process of a misuse detector and exploit the rule tree to deploy an online detector on low-resource equipment. A similar idea is pursued by Jacobs et al. [87] that introduces *TRUSTEE (TRUSt-oriented decision TreE Extraction)*, a framework for post-hoc global interpretability of black-box ML models applied in the domain of network security. The idea is to synthesize highly accurate and easily interpretable DTs starting from an ML model and its training dataset, along with a trust report that can be used to identify the components of the ML pipeline to be modified for improving the trustworthiness of the model. In the same vein, Meng et al. [168] present *Metis* which encompasses two methods to interpret DL-based networking systems. Metis utilizes teacher-student training to build DTs for local networking systems (e.g., congestion control agents on end-devices or flow schedulers on switches) and hypergraph formulations to generate interpretable policies for global networking systems (e.g., the controller in SDN).

Intrinsic/Explainable-by-design. Hereinafter, we describe explainable-by-design XAI techniques with intrinsic interpretability capability employed when performing NTA. Within this category, some works use transparent models or models that provide explainability information alongside their results, while others introduce innovative techniques that are inherently interpretable.

Concerning the first group of studies, various works exploit DT models to perform security-related NTA tasks. Mahbooba et al. [100] conduct a feature importance analysis based on information gain and analyze the rules that DT models follow when performing MD. Wang et al. [145] adopt a similar approach in their *TrafficAV*, which implements a scoring mechanism to analyze the underlying reasons behind malicious outcomes based on DT information gain. On the same line, Šarčević et al. [152] compare the "if-then" decision rules extracted from a DT with the SHAP outputs for tree models (i.e. the TreeSHAP feature attribution values) when performing AC. Dias et al. [131] propose an interpretable hybrid IDS that integrates rules crafted by experts with dynamic knowledge continuously generated by a DT as new pieces of evidence emerge from network activity. An interpretable IDS is also presented by Xu et al. [97]: it exploits an intrinsically-explainable additive tree for MD capitalizing on the features of normal and attack traffic biflows extracted via a shallow AE. We underline that the use of DT here differs from their application in post-hoc techniques, as in the former case such model is used as a surrogate explainer for a more complex model (i.e. tree-based techniques are not used for inference).

Another body of works proposes innovative explainable-by-design techniques for various NTA-related tasks. The work by Sejr et al. [127] focuses on detecting and explaining malicious HTTPS requests by implementing an unsupervised anomaly detector composed of three components: an n-gram vectorizer, a dimensionality reducer, and an outlier detector. A methodology for Distributed Denial of Service (DDos) attack detection based on a modified K-Nearest Neighbors (KNN) algorithm and a k-dimensional tree is devised by Feng and Li [149]. In the case a DDos attack is detected, the detector returns an alert message along with a risk profile representing the shortest distance to the legitimate traffic profile in the KNN searching space. Fauvel et al. [79] introduce a new *Lightweight, Efficient, and eXplainable-by-design convolutional neural network (LEXNet)* for Internet TC. LEXNet involves three steps: (i) the CNN backbone extracts discriminant features, (ii) the prototype block computes similarities to the learned class-specific prototypes, and (iii) the classification layer decides based on the computed similarities. The local explanations are given as application-specific prototypes stemming from the communication of detected prototypes and are compared with those obtained with post-hoc methods. In the context of incremental learning, Song et al. [75] propose an approach called I^2RNN *(Incremental and Interpretable Recurrent Neural Network*) for encrypted TC. I^2RNN is a modified version of a Long Short-Term Memory (LSTM) network and provides interpretability including time-series feature attribution (via feature ranking and identification of important ones) and inter-class similarity portrait. Minh et al. [134] present an approach involving the integration of multiple unsupervised models (base-learners), which are then combined through a stacking strategy. Notably, they use graphical representations of traffic flows to improve the explainability of the detection process. Each base-learner operates on pairs of features and the anomaly score is determined by the number of feature pairs deemed anomalous. This scoring method makes the results interpretable for security analysts since enables the identification of attack patterns based on visual cues.

Ge et al. [161] introduce *MetaCluster*, an interpretable classification framework that provides explainability by extracting feature prototypes at varying levels of granularity. This approach helps identify key patterns and semantics in network traffic that are easily interpretable for humans. While ongoing debates in the literature question the suitability of attention mechanisms as a measure of feature importance [41, 179], numerous studies utilize this technique to highlight the inputs that attract the model's attention, implicitly suggesting their significant contribution to the classifier's decision-making process. These studies regard both the ID [95, 150, 153, 176] and TC [80, 89] tasks.

3) Categorization by Model Dependency: Concerning the dependence of interpretability techniques on the model, we underline that most of the pre-model and post-hoc methods are **model agnostic**, while explainable-by-design methods are naturally **model specific**. For instance, all the works using DTs to design interpretable IDSs [97, 100, 131, 145, 152] base their explanations on peculiar rules/paths of tree-based models.

Nevertheless, some notable exceptions are worth to be underlined. In the case of post-hoc methods, model-specific ones all pertain to NTA tasks faced via CNNs and focus on specific aspects such as (i) filter activations for traffic fingerprinting [88], (ii) layer projections via feature maps for TC [68], and (iii) neuron activations for ID [147].

More sophisticated model-specific approaches are also proposed, such as GRACE [111] which combines K-means clustering with localization maps (Grad-CAM) for designing an interpretable CNN-based IDS. Another proposal tailored for the network-security domain is *DeepAID* [96], an interpretation method for unsupervised AD using DNNs. DeepAID encompasses a model-specific extension called *Distiller* that "distills" high-level heuristics from black-box DNN models and expert feedbacks into simplified finite-state machines.

4) Categorization by Explanations Types: Explanations in the NTA domain are provided in different forms depending on the specific interpretability technique used. In the following, we summarize illustrative use cases grouping them based on the categorization introduced in Section III-A.

Visual Explanations. Most NTA tasks faced using explainable CNNs [68, 88, 111, 147] provide explanations via different visual representations. Other works exploit model-agnostic graphical representations of data or other information related to the models they aim to elucidate. Many of these works leverage t-SNE [68, 76, 80, 147]. As notable examples, Beliard et al. [68] propose a platform to visualize the inference process of a commercial-grade network TC engine based on CNN that generates a set of interactive graphs (e.g., feature maps and t-SNE) to develop a better understanding of the most salient features driving the classification process. Furthermore, Wu et al. [147] use interactive visual analytics for interpreting and optimizing DL-based IDSs via feature maps, neuron activations, and t-SNE layer projections of CNNs. Ables et al. [106] also propose a SOM-based eXplainable IDS exploiting various visualizations, namely feature significance, U-matrices, and feature heatmaps. Analogously, Jeong et al. [133] provide an interactive web-based visualization system, designed using multiple incrementally-updated views for analyzing network traffic data via uncertainty quantification and discrete wavelet transform. The *Hybrid Oracle-Explainer* is devised in [94] to develop an IDS made of (i) an oracle encompassing a feature engineering module and a DNN and (ii) an explainer combining clustering results with DT-based visualization. The latter represents the DT corresponding to the closest centroid and whose prediction is the same as that of the oracle.

Input Importance. The prevalence of input-importance techniques is attributed to the specific interpretability needs in the networking field. Indeed, in practical NTA-related use cases, the urge is to obtain a deeper understanding of the relationship between the inputs and the resulting decisions [168, 180] rather than comprehending the internal workings of the model. As mentioned before, commonly post-hoc methods are also input-importance techniques working on an already-trained model. Among the studies proposing such methods, *EXPLAIN-IT* [165] represents a hybrid input-importance methodology for interpreting the results of unsupervised NTA tasks (e.g., analysis of Quality of Experience in YouTube video streaming). EXPLAIN-IT exploits LIME for interpreting the outcomes of a Support Vector Machine (SVM) traffic classifier trained on the results of an agglomerative hierarchical clustering: the most important features identified by LIME correspond to the characteristics responsible for the assignment to the clusters. Another input-importance technique proposed by Ahn et al. [175] utilizes a genetic algorithm to select the most crucial input features to optimize the trade-off between classification accuracy and model complexity (i.e. by removing unnecessary features) for service-specific TC. Other studies determine input importance through occlusion analysis: Rezaei et al. [67] investigate the impact of TLS extensions on the accuracy of DL models, Nascita et al. [139] evaluate the impact and potential bias of packet header fields (e.g., IP addresses and ports) on AC performance. Lastly, Nascita et al. [178] propose an innovative methodology for grasping the differences between traffic classifiers trained from scratch and incrementally, focusing on input importance (via SHAP), feature visualization, and analysis of base and incremental models.

Example-Based. Using specific instances for explanations is less common in the NTA domain. Nevertheless, an adversarial approach for IDS interpretation is proposed by Marino et al. [93]. Explanations are generated as minimum modifications of input features needed to correctly classify a given set of misclassified samples. Similarly, Burkart et al. [102] design an IDS complemented with explanations in the form of counterfactual examples for a MD task. Finally, Han et al. [96] leverage CADE that exploits contrastive learning to detect concept drift for individual samples deviating from existing classes and to explain the reasons behind the detected drift. CADE is evaluated in two NTA tasks: Android malware classification and general attack classification.

B. NTA Works on Reliability

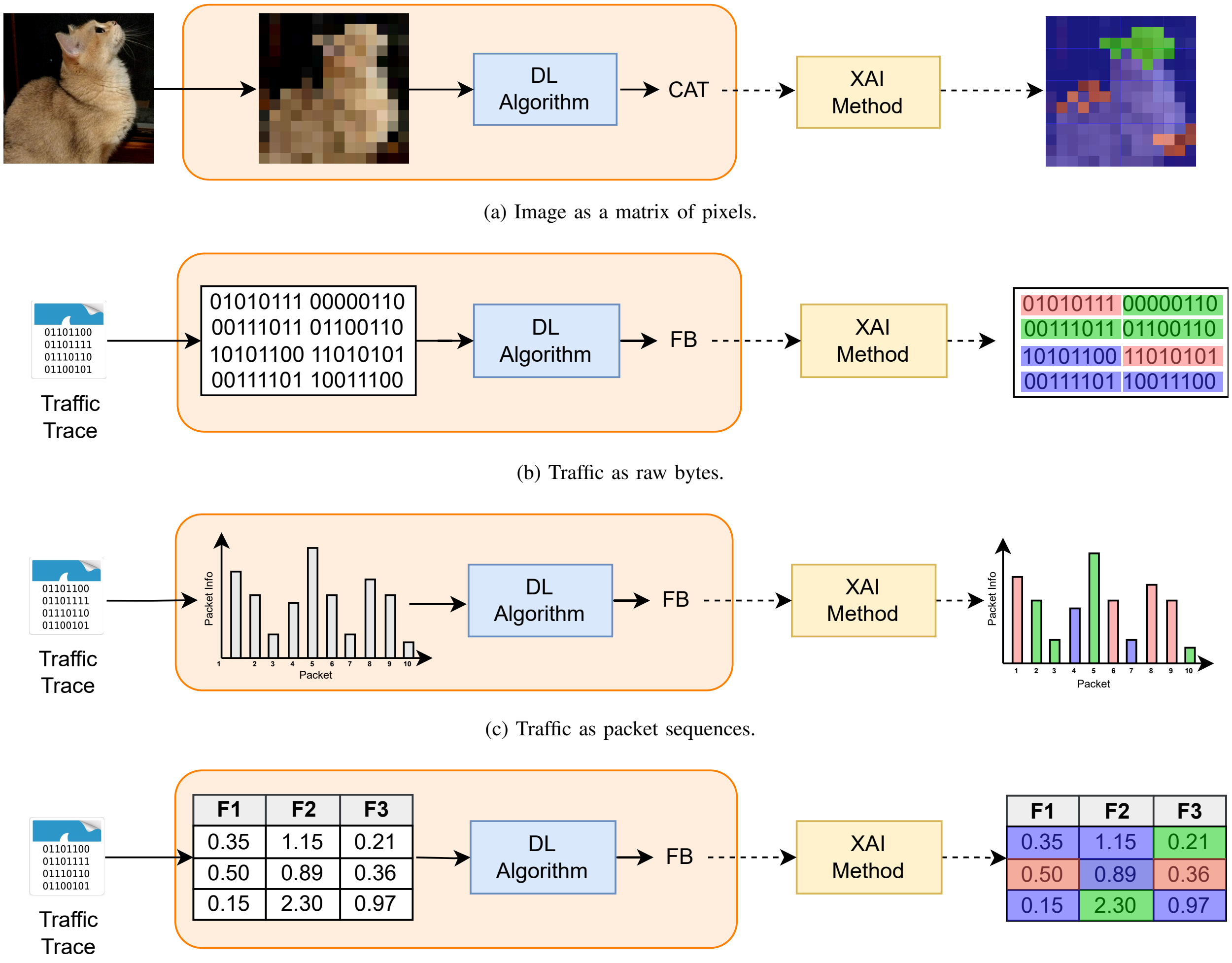
In this section, we delve into an examination of literature that investigates the reliability of tools used for the analysis of network traffic. Compared to the works that investigate the interpretability of the models, those that analyze their reliability are fewer in number and focus mainly on the calibration assessment and improvement of models trained for TC tasks [70, 71, 74, 77]. Differently, Li et al. [72] propose a trustworthy TC model which learns separately the classification predictions and the associated confidence scores. Specifically, a complementary neural network (*ConfidNet*) trained based on the probability of the true classes, is built to learn the confidence score of the classification model. A different approach is pursued by Jorgensen et al. [78] that incorporates techniques to quantify the uncertainty of predictions (i.e. a learned Mahalanobis distance) into the training process of a prototypical network [181]. Finally, the work by Guarino et al. [164] constitutes a first attempt to interpret a TP task by reducing the problem to a binary classification task and evaluating its calibration.

VII. From Network Traces to Input Data: the Importance of Representations

Defining the form to pass inputs to NTA tools is a critical step, which requires in-depth domain knowledge to design suitable traffic representations that are able to capture the important characteristics of the network traffic. This contrasts with the naive application to the computer vision domain, as stressed in Fig. 5. Indeed, network traffic is a complex entity, defined by the interaction among many distributed parties that communicate by exploiting heterogeneous media through a multitude of protocols that fulfill a large number of functionalities. The result is that a monitoring point placed in the middle of a network is traversed by *bidirectional streams of packets* that can be observed, analyzed, and interpreted at different extents based on the aggregation granularity. Based on such granularity, different TOs can be defined, thus determining how the captured raw packets are partitioned into distinct units with appropriate labels indicating the nature of the traffic (e.g., the benign or malicious application that injected it in the network).

Concerning studies focusing on the interpretability of AI-based NTA tools, commonly the network traffic is analyzed and managed being segmented in *flows*, representing a group of packets that share the same 5-tuple (consisting of the source IP and port, destination IP and port, and transport-layer protocol). It is worth noting that this definition takes into account the direction of the traffic. Nevertheless, in many studies, flow direction is disregarded in the traffic segmentation phase, and all packets from both directions are treated as part of a single flow, often referred to as a *biflow* [169].

Once the TO is defined, experts in charge of designing NTA tools have further knobs at their disposal, as different choices can be made in terms of input-data representation. It is worth noting that the definition of the TO is crucial, as the representations that can be designed may depend on such choice.



(a) Image as a matrix of pixels.

(b) Traffic as raw bytes.

(c) Traffic as packet sequences.

(d) Traffic as pre-extracted features.

Figure 5. Relation between explainability and input representation in image classification (a) and app traffic classification (b–d). Colors map to the importance of the portions of the input (red=negative importance, blue=no importance, green=positive importance). Different input representations result in different degrees of explainability. Image representation is not subjected to discussion and is naturally understandable by humans (a). Hence, the role of the input portions is naturally explainable with no ambiguities. On the other hand, the input-design choices available for network traffic (raw bytes (b), packet sequences (c), pre-extracted features(d)) are intuitive to a human operator at different degrees and impact on the interpretability of the explanations.

Notably, in other domains such as computer vision, the representation of the information is not subjected to discussion (i.e. an image is a two-dimensional array of pixels). When referring to such representation, the role of the input (i.e. how its portions contribute to the decisions of the model) is naturally explainable with no ambiguities (see Figure 5a). Referring to a widely-known example from the computer vision domain, highlighting the portion of the image that led the classifier to output “husky dog” allows one to understand whether this choice was erroneously dictated by the presence of some snow in the background. Indeed, as the portion of the input (matrix of pixels) is directly understandable by humans, when an image area is highlighted as being more important there is no effort or specific skill required to understand the meaning: the semantic gap is minimal. Differently, when dealing with NTA the different input-design choices available (related to TO segmentation and data representation) impact the interpretability of the results to different extents. In this section, we delve into the types of inputs that are used in works applying XAI techniques to NTA.

A. *Raw Bytes*

In order to fully capitalize DL end-to-end learning capability, many approaches feed the traffic-analysis pipeline directly with raw-byte sequences exchanged across the network (see Figure 5b). In general terms, raw bytes can be extracted in different forms, e.g., considering the payload of L2, L3, or L4 Protocol Data Units. The specific choice is related to the specific application domains. For instance, when focusing on AC or AD, considering the content of even the header of L2 frames can be highly informative. On the other hand, when the aim is classifying app traffic, the bytes constituting the L5 messages are usually in the spotlight.

Commonly, studies employing raw-byte sequences undertake preprocessing procedures aimed at eliminating privacy-sensitive details or those introducing unwanted bias possibly causing overfitting, e.g., being closely linked to the particular network environment where data are gathered. Such details may include header fields such as IP and MAC addresses or Server Name Indication (SNI).

For example, Garcia et al. [73] employ the first 1500 bytes of each packet. To prevent overfitting, a preprocessing phase is implemented, encompassing various packet transformations that include removing the Ethernet header, masking IP addresses, adding padding to UDP headers, and removing irrelevant packets. Luo et al. [76] employ payload bytes with IP addresses and ports removed and padding of UDP header to obtain data alignment of TCP header and UDP header. Their approach considers a maximum number of bytes set to 1500, converts each element of the byte array to binary, and then re-encodes to integers by grouping them in groups of 1 (i.e. binary form), 2, 4 or 8 bits (i.e. decimal form). Wang et al. [69] leverage the first 1014 transport-layer bytes of each biflow, after removing all the packet headers (i.e. Ethernet, IP, and TCP/UDP). Wang et al. [89] let the number of bytes of the first IP packet of each flow vary in the range $\{16, 32, 64, 128\}$ and select the dimension by considering the best trade-off between the classification accuracy and time overhead on validation data. On the other hand, Sejr et al. [127] consider the byte sequence corresponding to HTTP requests in its plaintext form.

As an alternative approach, certain studies employ an additional representation step before sending bytes across the classification system. For instance, Li et al. [72] take the first 784 L4-payload bytes and convert them into images (with dimensions of 28×28), while Luis-Bisbé et al. [86] consider the first 1024 L4-payload bytes and convert them into images (with dimensions of 32×32). Notice that obtaining bidimensional representations from monodimensional byte sequences typically is not soundly motivated and represents an obstacle to the practical interpretation of the outcome of the analysis by itself.

Notably, the impact of content encryption plays a major role when dealing with such representation. On the one hand, encryption heavily impairs the performance of NTA tools relying on recurring signatures in packet content. On the other hand, it also prevents the interpretation of the results that can be hardly put in relation with input data (differently than what occurs when dealing with sequences of fields exchanged in clear). In this sense, via an occlusion analysis, Bovenzi et al. [182] assess the robustness of the proposed classification solution against encryption mechanisms that compromise specific portions of the input (TLS SNI) at different extents.

Finally, it is worth noting that the majority of the papers that employ bytes as data representation primarily address TC.

B. Packet Sequences

Several studies have explored the adoption of informative fields extracted from a sequence of packets modeled as a time series (see Figure 5c). These informative fields include the packet Direction (DIR), the Inter Arrival Time (IAT) (i.e. the time between consecutive packet), the TCP Window Size (TCP WS) (i.e. the dimension of the TCP window), the Payload Length (PL) (i.e. the number of bytes in the transport-layer payload), the overall Packet Size (PS), the Time To Live (TTL), and the TCP flags.

Since the adoption of these informative fields allows for avoiding the inspection of the content of the packets, this strategy is convenient from several points of view. First, the extraction process is more lightweight compared to Deep Packet Inspection, thus supporting faster input extraction (even at line speed). Secondly, these sequences are practically less impacted by encryption, thus proving more robust.

The difference among papers utilizing this data type lies in the specific fields they employ and the number of packets they refer to. For instance, Nascita et al. [71, 77] leverage DIR, IAT, TCP WS, and PL considering a varying number of packets. Yan et al. [101] employ the sequence of PS for the first 9 packets. Song et al. [75] leverage PL, DIR, IAT, TCP WS, TCP flags, and TTL, adjusting the number of packets based on the application whose fingerprint is calculated. Fauvel et al. [79] and Beliard et al. [68] utilize PS and DIR from the first 10 and 20 packets, respectively. Luxemburk and Čejka [83] consider IAT, DIR, and PS for the first 30 packets. Guarino et al. [164] employ 10 packets and for each of them extract DIR, IAT, and PL.

Based on the studies reviewed in this survey, it is apparent that this input type is predominantly employed for TC, and it is also complemented with raw input data and pre-extracted features.

C. (Pre-extracted) Features

A number of works [81, 82, 85, 92–100, 102–126, 128–131, 133–138, 140–156, 158–161, 165, 166, 175, 177] design NTA approaches that are fed with preprocessed data representing TOs via a set of concise features whose definition is usually guided by domain experts and are possibly *handcrafted*. In this case, network traffic is generically represented as tabular data, with each row representing a TO and each column representing a feature (see Figure 5d).

In more detail, several of the surveyed papers employ statistical features recorded via *NetFlow*[1] [104, 109, 118, 119, 126, 136, 143, 149] or generated from the collected network traffic, e.g., by using tools such as *CICFlowMeter*[2] [94, 96, 102, 103, 106, 109, 110, 113, 128–130, 133, 134, 142, 151, 154, 155, 159]. Specifically, CICFlowMeter allows for the calculation of over 80 statistical network traffic features related to the entire (bi)flow. These include either features associated with the whole biflow lifetime (e.g., duration, number of packets, number of bytes) or features obtained by summarizing per-packet properties (e.g., packet or header length). In the latter case, generally, the minimum, the maximum, the mean, and the standard deviation are computed. A similar reasoning also applies to counts of packets having specific TCP flags set (e.g., FIN, SYN, RST, PSH, ACK, URG, CWR, and ECE).

On the other hand, other works extract different sets of features defined *ad hoc*. For instance, Ahn et al. [175] extract 20 statistical features derived from the sequences of PSs and IATs (i.e. minimum, maximum, average, and standard deviation), along with the number of packets and bytes for each flow. Kundu et al. [138] extract 199 different features from packet headers after performing traffic aggregation into

[1] https://datatracker.ietf.org/doc/html/rfc3954.

[2] https://github.com/CanadianInstituteForCybersecurity/CICFlowMeter.

unidirectional and bidirectional flows. These features are categorized into four groups: aggregated, temporal, statistical, and connection-context features. Hsupeng et al. [155] add a set of features obtained through their *CSTITool* to the common features extracted with CICFlowMeter. Other expert-driven features describing byte distribution and incorporating domain-specific knowledge are also taken into account. Wang et al. [145] leverage 4 features from HTTP requests and 6 from TCP flows, while Chowdhury et al. [81] uses 216 features related to packet, protocol, rate, stochastic and time information. Dias et al. [131] exploit 5 features: destination IP, destination port, protocol, PS, and packet timestamp.

Caforio et al. [111], Andresini et al. [150], and Wu et al. [147] follow a different approach and transform the features into images before providing them to their models thus reformulating the network TC task as an image classification problem.

When using pre-extracted features, different datasets likely provide different sets of features, thus introducing severe challenges in assessing the robustness of models when applied to different network scenarios. Furthermore, pre-extracted features are typically computed using *post-mortem* (viz. *offline*) approaches. Indeed, although it is possible to extract such features by observing only a certain number of packets, the most common method is calculating them after observing the whole biflows. For this reason, utilizing such an input type is usually not associated with *quick verdicts* (e.g., *early NTA* [70]). On the other side, relying on pre-engineered and human-understandable features may positively impact in terms of attainable interpretability, as the results are naturally related to input data that are understandable by humans.

D. Heterogeneous Inputs

Notably, the majority of proposals investigating XAI for NTA focuses on strategies relying on a single type of input, thereby employing single-modal approaches. On the other hand, some other studies use multiple input types, therefore aiming at interpreting the role of each in generating the output.

Specifically, Luxemburk and Čejka [83] and Luxemburk et al. [84] exploit packet sequences (IAT, DIR, and PS) in conjunction with per-flow statistics, while the works by Nascita et al. [71] [77] leverage payload bytes and packet sequences (IAT, PS, DIR, and TCP WS). The paper by Guarino et al. [74] is the only one employing all three input types together.

When dealing with heterogeneous inputs, the interpretability can be achieved at different levels. For instance, Nascita et al. [71, 77] first aim at defining which is the role of each modality (i.e. a set of inputs) in determining the results, and then the specific modality is further dissected. Differently, Luxemburk and Čejka [83] and Luxemburk et al. [84] only focus on each modality individually.

E. Considerations on Inputs and Impact on Explainability

Table VI provides a quantitative summary of the input types as adopted by works that focus on XAI for NTA. The analysis of the literature witnesses that most of the studies take advantage of concise features as they leverage the effort of other researchers who have previously collected and preprocessed the data, thus providing a set of features ready to be fed to NTA models. This decision is motivated by the desire to ease the workflow to expedite model training and evaluation. Additionally, it is justified by the fact that in many cases raw data are not readily available for researchers to extract alternative representations or features of their definition.

Table VI
COUNT OF THE WORKS USING THE DIFFERENT TYPES OF INPUT, DIVIDED BY TASK AND IN TOTAL.

	Task					
Input	**TC**	**ID**	**AC**	**TP**	**Other**	**Total**
Only Raw Bytes	9	1	1	0	0	11
Only Pkt Sequence	4	1	1	2	0	8
Only Features	6	43	32	0	4	85
Raw Bytes, Pkt Sequence	3	0	0	0	0	3
Raw Bytes, Features	1	1	1	0	0	3
Features, Pkt Sequence	2	0	0	0	0	2
Raw, Pkt Sequence, Features	1	0	0	0	0	1

Whatever the cause behind input definition, it influences the choice of XAI techniques that can be used and has a direct impact on the nature of the explanations related to the role of the inputs and their usability. This is evident when input importance techniques are applied, because the explanations are defined in terms of the model inputs (cf. Section III). In practice, when applying XAI techniques to expert-driven features extracted from traffic data (which have a well-defined meaning), **it becomes easier and more immediate to reason about the obtained explanations and speculate on their reliability**.

On the other hand, **when working with raw bytes, achieving interpretability of the results becomes more challenging**. Despite this difficulty, the adoption of interpretability techniques is even more crucial in this case, as this input representation is the least intuitive among those described. Therefore, it is essential to shed light on how (and how much) the portions of the input contribute to the decisions of the models. Notably, when considering other application domains such as computer vision, DL models naturally process images and input contribution is more easily understandable by human operators. For instance, explanations generated by XAI techniques can be presented using heatmaps, which highlight how the various regions within the images contribute to the predictions (see Figure 5a). This approach is intuitive for humans, aiding in understanding why a model makes a particular choice.

In principle, this approach could also be pursued with network traffic, but it is certainly much more complicated for the end user to benefit from the final explanation than it is for images. Unlike the latter, where specific regions can move around but still convey the same information, **network traffic requires expert interpretation to identify critical bytes, sequences, or protocol fields**. The widespread use of encryption further complicates this analysis, making it difficult or even impossible to extract meaningful insights.

Another difference to highlight with respect to the image domain is that the **representations we can obtain for network traffic are more structured than images**. In the latter case, a specific subpart of the image can be in different positions within the same image and, therefore, the same visual information may contribute similarly to the output despite being in different positions. This causes explanation aggregations to lose meaning and makes local explanations more suitable for these input types. On the contrary, **some of the representations extracted from traffic lend themselves to other types of aggregations**—in addition to the spatial aggregation discussed in Section VI-A—**to obtain different views and different insights on the problem** (e.g., with packet sequences input we can aggregate considering protocols, specific fields, or packets).

Ultimately, the differences between computer vision and networking highlight why XAI struggles with NTA. **The weaker semantic content in networking data limits the effectiveness of XAI, indicating the need for specialized approaches tailored to the unique demands of NTA**.

VIII. Datasets, Libraries, and Tools: the road towards the reproducibility

This section discusses two fundamental aspects underpinning the reproducibility of XAI-based analysis in the NTA domain: public datasets (Section VIII-A) and libraries and tools available (Section VIII-B).

A. Datasets

Having high-quality and publicly available traffic datasets represents an invaluable resource, fostering reproducibility and thorough evaluations and fueling XAI research to support NTA. Sadly, data availability is a primary concern in this domain. Indeed, standardized procedures for generating, gathering, preprocessing, labeling, and disseminating traffic data are indispensable. However, these methods are frequently absent, primarily due to diverse and privacy-sensitive collection contexts. This notwithstanding, the scientific community has provided tangible effort in releasing traffic that has been used or can be used for studies related to NTA.

We collect, analyze, and categorize these datasets, which are summarized in Table VII.

The table presents 51 datasets and includes both the datasets already leveraged to support XAI studies and examples of those we believe are valuable to be considered by researchers for future studies. For each dataset, we provide details to guide their adoption for future works. The pieces of information reported include the capture span and the release year of the dataset, as well as the nature of collected traffic (e.g., mobile apps, desktop, IoT, security, and anonymity tools) together with the related label space and the modality the traffic capture was conducted (either automatically or human-generated). In addition, the table also highlights whether raw traffic traces are available (which guarantees higher flexibility in data representation). Finally, we indicate which datasets were used in the works collected in this survey and, in more detail, whether the scientific community has conducted explainability analyses concerning the NTA tasks discussed in Section IV (i.e. TC, AC, ID, TP). This represents another important aspect of reviewed works and can assist researchers in understanding and comparing the results of their analyses.

Inspecting Table VII, it is evident that the majority of the datasets lie in the security domain (30/49), with the other domains being poorly represented in most of the cases. Interestingly, only a limited number of datasets (14) rely (completely or partially) on human-generated captures. However, in most cases, raw data are provided, allowing for more freedom in data representation and supporting a wide range of XAI methodologies. Finally, focusing on the four rightmost columns, our analysis highlights how XAI studies have mostly posed their attention on a larger number of different datasets for legit TC, with ID and AC being also in the spotlight, witnessing the recent attention of the scientific community to network security issues.

To conclude, we believe that although being limited and requiring additional effort to be consolidated, the endeavor of the scientific community in providing shared datasets to benchmark and reproduce results is evident and represents an important resource to support XAI studies in the NTA domain in the future.

B. Libraries and Tools

Beyond quality datasets, ready-to-use implementations of XAI methods are the other side of the coin that allows existing studies to be consistently reproduced. Fortunately, the scientific community can leverage plenty of (open-source) tools and coding libraries allowing for running XAI methods in the NTA domain with minor implementation effort for their adaptation. In Table VIII, we list the most popular libraries/tools for explainability analyses. For each, we report the related release/update timing information and the XAI techniques covered. Moreover, we highlight the ones providing metrics for evaluating the explanations from different perspectives, when available.

Remarkably, most of the reported tools were released after 2018, underlining the recent and increasing interest of the scientific community in this topic in the last 5 years, which is further witnessed by the continuous updates of the software (continuous revisions are available, with the majority of the projects being updated in the last two years).

Some libraries reported in the table implement a single technique and are often maintained by the authors of the paper proposing the technique (e.g., SHAP [48], LIME [49], Anchor [223], Weight Watcher [234]). On the other hand, other libraries gather a number of known techniques presented in the scientific literature (e.g., AIX360 [221], Xplique [233], OmniXAI [230], Alibi [222]). Such aggregators are particularly useful for benchmarking analyses (i.e., when different XAI techniques are to be compared) since they provide a common interface for different methods and help mitigate the issues encountered in integrating different libraries with diverse dependencies. Notably, *all* the entries listed in the table have Python as their programming language, witnessing its wide adoption and versatility in the field of data science.

The majority of libraries/tools provide explainability techniques in the strict sense, while others serve as tools to gain

Table VII
PUBLIC DATASETS EMPLOYED FOR NTA. LISTED BY RELEASE YEAR (↓). THE NAMES OF DATASETS ARE CLICKABLE LINKS, DIRECTING TO THEIR RESPECTIVE REFERENCE PAGES.

	Release	Traffic			Raw		XAI applied for			
Dataset	Year↓	Nature	🧍	Label Space	Data	Capture Span	TC	AC	ID	TP
KDDCup99 [183]	1999	🔒	○	4 attacks categories, 38 attacks		N/A		✓	✓	
Discovery Challenge ECML/PKDD [184]	2007	🔒	○	normal & 7 attacks		N/A			✓	
CAIDA 2007 DDoS*	2007	🔒	○	ICMP flood attack	✓	04/08/07 (1h)		✓		
NSL-KDD [185]	2009	🔒	○	4 attacks categories, 38 attacks		N/A		✓	✓	
DARPA 2009 DDoS*	2009	🔒	○	TCP SYN flood attack		05/11/2009		✓		
UniBs [186]	2011	🔒	○	7 Traffic Types	✓	10/2009 – 11/2009	✓			
FRGP NTP Flow Data*	2014	🔒	○	NTP reflection attack		01/12/2013 – 28/02/2014		✓		
UNSW-NB15 [187]	2015	🔒	○	benign / 9 attack types	✓	22/01/15 & 15/02/2015		✓	✓	
UPC [188]	2015	🔒	○	17 protocols, 25 apps, 35 web services	✓	25/02/13 – 01/05/13		✓		
DDoS Chargen 2016*	2016	🔒	○	UDP reflection and amplification attacks		25/11/2106 – 26/11/2016		✓		
UGR'16 [189]	2016	🔒	○	normal & 10 attacks		4 months			✓	
ISCXVPN2016 [190]	2016	🖥	●	2 encaps. types / 7 traffic types / 15 apps	✓	03/15 – 06/15	✓			
ISCXTor2016 [191]	2016	🕵	●	8 traffic types / 18 apps	✓	07/15 – 02/16	✓			
USTC-TFC2016 [192]	2017	🔒	○	benign (10 apps) / 10 Malware types	✓	2011 – 2015	✓			
CIC-IDS2017 [193]	2017	🔒	○	7 attacks	✓	03/07/17 – 07/07/17		✓	✓	
KITSUNE [194]	2018	🔒	○	4 attack types / 9 attacks	✓	N/A		✓	✓	
UNSW(IoT) [195]	2018	🤖	●	28 devices	✓	10/16 – 04/17	✓			
CSE-CIC-IDS2018 on AWS [193]	2018	🔒	○	7 attacks	✓	02/18 – 03/18		✓	✓	
Mon(IoT)r [196]	2019	🤖	○	26 IoT devices	✓	85 days	✓			
Cross-Platform-IOS [197]	2019	📱	○	185 apps		N/A	✓			
Cross-Platform-Android [197]	2019	📱	○	167 apps		N/A	✓			
Mirage-2019 [198]	2019	📱	●	40 apps		05/17 – 05/19	✓			✓
CICIDDoS2019 [199]	2019	🔒	○	13 attacks	✓	12/01/19 & 11/03/19		✓		
BoT-IoT [200]	2019	🔒	○	benign / 10 attacks	✓	N/A		✓	✓	
ToN-IoT [201]	2019	🔒	○	benign / 9 attacks		N/A		✓	✓	
Mirage-Video [163]	2020	📱	●	4 video categories / 8 apps		06/19 – 03/20				✓
IoTID20 [202]	2020	🤖 🔒	○	normal, 4 attack types, 8 attacks	✓	N/A		✓	✓	
CIC-DARKNET2020 [203]	2020	🕵 🖥	●	benign & darknet, 8 traffic types	✓	03/15 - 06/15 & 07/15 - 02/16	✓			
WUSTL-IIoT	2021	🤖	○	normal & 4 attack categories		N/A		✓	✓	
USB-IDS-1 [204]	2021	🔒	○	benign / 16 attacks		N/A			✓	
RegSOC–KES2021 [99]	2021	🔒	○	normal & anomaly		N/A			✓	
Mirage-Covid-CCMA-2022 [205]	2022	📱	●	9 apps / 3 user activities		04/21 – 12/21	✓			
AppClassNet [206]	2022	📱 🖥	●	500 apps		N/A	✓			
CESNET-QUIC22 [207]	2022	📱 🖥	●	7 traffic types / 102 services		31/10/2022 – 27/11/2022	✓			
CESNET-TLS22 [83]	2022	📱 🖥	●	200 services		10/11 (2 weeks)	✓			
IoT23	2022	🔒	○	benign / 9 attacks	✓	2018-2019		✓		
IEC 60870-5-104 ID [208]	2022	🔒	○	14 attacks	✓	N/A		✓		
VPN/nonVPN Network App Traffic [78]	2023	🖥	○	5 categories, 10 apps	✓	06/19 – 06/20	✓			
							No XAI application			
ISCXIDS2012 [209]	2012	🔒	○	benign / 4 attacks	✓	06/10 (1 week)				
Anon17 [210]	2017	🕵	●	3 anon. tools / 8 traffic types / 21 apps		2014 – 2017				
MTD [211]	2018	📱	●	12 apps		10/16 – 03/17				
QUIC [212]	2018	🖥	○	5 QUIC services		03/18				
N-BAIoT [213]	2019	🔒	○	benign / 10 attacks		N/A				
IoT Network Intrusion Dataset	2019	🔒	○	benign / 9 attacks	✓	N/A				
MQTT-IDS-IoT [214]	2020	🔒	○	benign / 4 attacks	✓	N/A				
Orange'20 [215]	2020	📱	●	8 traffic types		11/07/19				
UTMobileNetTraffic2021 [216]	2021	📱	◐	16 apps / 31 user activities		03/18 – 04/18				
CICIoT2022 [217]	2022	🤖 🔒	◐	3 device types / 40 devices / 2 Attacks Types	✓	09/21 – 12/21				
ETF-IoT-Botnet [218]	2022	🔒	○	6 attacks	✓	2019-2021				
EDGE-IIOT [219]	2022	🔒	○	5 attack types / 14 attacks	✓	11/21 – 01/22				
CICIoT2023 [220]	2023	🔒	○	7 attack types, 33 Attacks	✓	N/A				

Traffic Nature: 📱 = Mobile Apps, 🕵 = Anonymity Tools, 🖥 = Desktop, 🤖 = IoT, 🔒= Security.
🧍 = Human-generated: whether it is completely/partially generated by real human experimenters, instead of bots or scripts.
Raw Data: PCAP files are available.
*The datasets marked with an asterisk include traffic from a single class of attack. To effectively conduct AC tasks, it is essential to merge them into one dataset, following the approach outlined in the work [149].

Table VIII
TOOLS ORGANIZED IN ALPHABETICAL ORDER (↓). THE NAMES OF TOOLS/LIBRARIES ARE CLICKABLE LINKS, DIRECTING TO THEIR RESPECTIVE REPOSITORIES. (LAST ACCESS TO SOFTWARE REPOSITORIES: JAN 2024)

Tool/Library ↓		Commit Yr. First-Last	XAI techniques
AIX360 [221]	♦	2019–23	BRCG, Generalized Linear Rule Models, ProtoDash, ProfWeight, Teaching Explanation for Decision, Contrastive Explanations Method, CEM with Monotonic Attribute Functions, Disentagled Inferred Prior Variational Autoencoder
Alibi [222]		2019–24	Accumulated Local Effects, Anchors, Counterfactual Instances, Contrastive Explanation Methos, Counterfactuals Guided by Prototypes, IG, SHAP
Anchor [223]		2018–22	Anchor
Captum [224]	★	2019–24	Grad-CAM, GuidedBackProp, IG, DeconvNet, SHAP, Occlusion
Dalex [225]		2018–23	Partial Dependence Plot (PDP), LIME, Accumulated Local Effects Plot, Merging Path Plot, Shapley Values
DiCE [226]		2019–23	Counterfactual Explanations
DoWhy [227]		2018–24	Effect Estimation, Quantify Causal Inferences, What-if analysis, Root cause analysis and explanations
Dtreeviz		2018–24	DT Visualization
ELI5		2016–20	LIME, Permutation Importance, Grad-CAM, TextExplainer
explainX		2020–24	SHAP, What-if analysis, Model Performance Comparison, PDP
ExplainerDashboar		2019–23	PDP, SHAP, Shap interaction values, Permutation Importance, Visualization of RF Trees
H2O		2017–20	Shapley Feature Importance, Feature Importance, PDP, Individual Conditional Expectation (ICE), DT, Local Linear Explanations, Global Interpretable Model
InterpretML [228]		2019–24	PDP, Explainable Boosting, DT, Decision Rule List, Linear/Logistic Regression, SHAP, LIME, Morris Sensitivity Analysis
iNNvestigate [229]		2017–23	PDP, Perturbation Analysis, Gradient*Input, SmoothGrad, IG, DeconvNet, Guided BackProp, PatternNet, LRP, Shapley Value Sampling
LIME [49]		2016–21	LIME
OmniXAI [230]		2022–23	Grad-CAM, Grad-CAM++, Score-CAM, LayerCAM, PDP, GuidedBackProp, IG, Accumulated Local Effects, Sensitivity Analysis, Counterfactual Expl, Contrastive Expl, SHAP, LIME, SmoothGrad, Learning to Explain
PyCaret		2019–24	Calibration Curve, Feature Importance, t-SNE, SHAP, PDP, Morris Sensitivity Analysis, PFI
Py-CIU		2020–24	Contextual Importance and Utility
SHAP [48]		2016–24	SHAP
Shapash	■	2020–24	SHAP, LIME
Skater		2017–23	PDP, LIME, IG, Feature Importance, LRP, Tree Surrogates, Scalable Bayesian Rule Lists
Tensorboard		2015–24	Visualization Techniques
Tf-explain		2019–22	Saliency Maps, Activations Visualization, Vanilla Gradients, Graident*Inputs, Occlusion Sensitivity, Grad-CAM, SmoothGrad, IG
TSViz [231]		2019–19	Visualization Techniques for Time-Series Analysis
Weight Watcher		2018–24	Diagnostics Techniques (layer-by-layer) for DL models
What-If Tool [232]		2018–23	Visualization, Probe, Interactive Evaluation
XAI		2019–21	Imbalance Analysis and Mitigation, Feature Correlation, Permutation Feature Importance (check all)
Xplique [233]		2020–23	SHAP, LIME, Occlusion, Rise, Sobol Attribution, Hsic Attribution, DeconvNet, Grad-CAM, Grad-CAM++, GradientInput, GuidedBackPropegation, IG, Saliency, SmoothGrad, SquareGrad, VarGrad
Yellowbrick		2016–23	Visualization Techniques

Explanation-quality metrics:
We report the metric name as used in the tool documentation, and within parentheses the corresponding property as systematized in [4].
♦: Faithfulness, Monotonicity (both ∈ Correctness)
★: Infidelity, Sensitivity (both ∈ Continuity)
■: Stability (Continuity), Consistency (Consistency), Compacity (Completeness)

insights into models and analyze data. Often, these tools enable the extraction of specific visualizations concerning various aspects of the models under examination, thereby enhancing the understanding of the model itself. Beyond those listed in the table, it is worth mentioning some other libraries that aim to investigate/evaluate fairness and bias mitigation. Among these, we find AIF360 [235], Aequitas [236], Fairlearn [237], Fat-Forensics [238], PyCaret [239].

Concerning the metrics, although they represent a crucial aspect for reaching *good* explanations, it is evident from the table that only a limited number of tools (3) provide implementations of metrics for evaluating the explanations. Furthermore, we remark that there is no agreement on the nomenclature of metrics nor on their definition (see Section V-A). This does not make it straightforward to understand which metric is available in each library and which aspects it pertains to. To avoid ambiguities, in the footnote of Table VIII we report the metrics with the names used in the respective libraries and match them with the categories systematized in [4].

Last but not least, another aspect that should be carefully considered when choosing the library to leverage regards the cost related to an explainability technique. However, such an evaluation requires several factors to be taken into account: the theoretical complexity of the technique, the language used, and the specific implementation provided within a particular tool or library. Given the complexity of such evaluation and the current maturity of XAI field, it is unsurprising that current libraries overlook this aspect.

The two aspects discussed above, concerning metrics and complexity, represent significant challenges that must be addressed to develop more comprehensive frameworks for evaluating XAI techniques. Tackling these challenges is not only essential for enhancing the comprehensiveness of the frameworks but it is also crucial for achieving full reproducibility of results that take these aspects into account.

IX. Concluding Remarks on Open Challenges

Integrating the evolving landscape of XAI into NTA poses both promising advancements and formidable challenges. While XAI holds the potential to enhance our understanding of network behaviors and facilitate more informed decision-making processes, its application in this domain confronts several complex hurdles [240]. These challenges range from leveraging XAI at the design stage for improving and adapting NTA tools to the inherent trade-offs between accuracy and transparency and the unavoidable costs of interpretability as well [2]. Additionally, providing NTA-focused XAI tools and ensuring the robustness via XAI frameworks in dynamic network environments remains a critical concern.

In this section, we discuss the *open challenges and gaps* that arise when applying XAI techniques to NTA, shedding light on the areas that demand further investigation and innovation to realize its full potential in the product line.

A. Inadequate Methods for XAI in the Loop

The integration of XAI methods into the decision-making loop is essential for **achieving actionable improvements in close-to-automatic fashion**. XAI methods should not only elucidate the reasons behind predictions but also explicitly indicate the necessary steps to enhance the quality of these predictions. In addition, future XAI methods should embrace a holistic approach by seamlessly integrating with cutting-edge tools or platforms (e.g., ModelOps) to support this process. By doing so, they can extract deeper insights from the data and effectively associate actionable recommendations with decision-making processes. This integration would enable a more productive and data-driven decision-making ecosystem, where XAI serves as a key enabler of continuous improvement and informed choices.

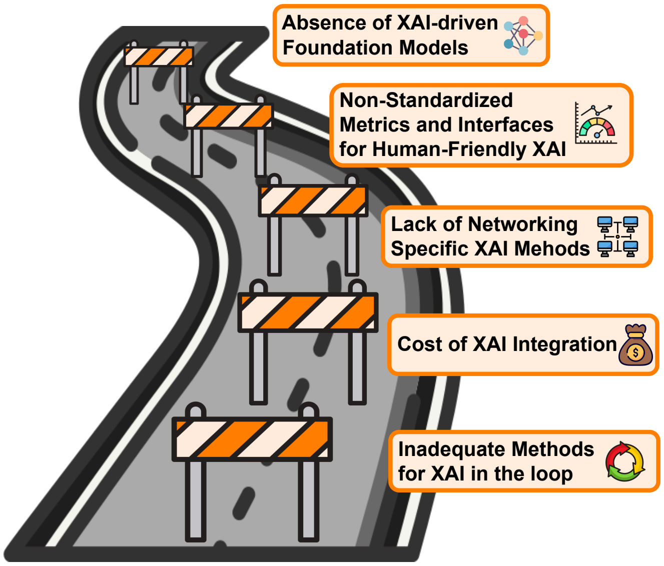

Absence of XAI-driven Foundation Models
Non-Standardized Metrics and Interfaces for Human-Friendly XAI
Lack of Networking Specific XAI Mehods
Cost of XAI Integration
Inadequate Methods for XAI in the loop

Figure 6. Open Challenges for XAI in Networking.

Further, given the dynamic nature of network traffic, NTA models **evolving via lifelong learning techniques are a key challenge**. In such a case, the need for explainability is further heightened. In fact, incremental training is inherently more complex with resulting models exhibiting subpar performance. Hence, explainability is critical to enable a comprehensive analysis of the model's internal dynamics, understand the factors contributing to poor performance, guide diagnosing, make informed adjustments, uncover the hidden challenges within the model, and rectify its shortcomings. In this context, XAI methods encounter challenges due to the **lack of methodologies tailored for incremental training**. Indeed, the dynamic nature of incremental updates surpasses conventional XAI tools' capabilities, hindering a comprehensive understanding of model decisions over time. Addressing this gap requires the development of robust and standardized approaches that would enable meaningful insights into evolving models' decision-making processes, enhancing our understanding of their behavior.

B. Cost of XAI Integration in NTA-based Systems

A first observation regarding the integration of NTA systems into XAI is the need for carefully taking into account the **accuracy-interpretability trade-off**. The latter problem refers to the inherent conflict between achieving high predictive accuracy and maintaining interpretability (or comprehensibility) in predictive models or algorithms within the field of ML and data analysis.

Achieving interpretability through XAI also entails **associated computational cost** [241]. Additionally, **economic factors** [242] must be considered when embracing XAI into NTA.

One challenge with current XAI solutions is their difficulty in integrating into networking systems for **on-the-fly model interpretation**. To enhance AI confidence within the networking community, there is a growing need for system-level support. This includes the development of standard application programming interfaces and software development kits aimed at seamlessly incorporating XAI techniques into the operational network environment. These advancements would enable real-time, automated scrutiny and validation of various AI-based solutions.

From an economic perspective, **organizations should consider the specific requirements of their application, available resources, and constraints** to determine the feasibility of implementing XAI techniques in their NTA systems. This involves also considering the costs of creating and storing audit logs, the impact on innovation speed (e.g., time-to-market for network devices), and the potential loss of flexibility due to future shifts that may not align with prior explanations.

C. Lack of Specialized XAI Methods in Networking

Previous sections have underlined that existing studies commonly utilize XAI methods introduced in other application domains like SHAP and LIME. Such methods are **not inherently crafted to harness the distinctive features of contemporary networking systems and data** (see Section VII-E). Therefore, this approach may result in inconsistent or misleading outcomes. Accordingly, it is crucial to account for the specificity of the target problem and develop tailor-made XAI methods that align with the corresponding network and system configurations. One relevant example is the integration of causal explanation methods aligning to the aforementioned constraints. As modern networks grow in complexity, there is a need for the creation of more XAI techniques specifically tailored for current NTA tools and network settings.

D. Non-Standardized Metrics and Interfaces for Human-Friendly and Trusted XAI

Stakeholders in NTA include network administrators, network service providers, cybersecurity analysts, regulatory authorities, and end-users. All these categories can benefit from the provision of explainability, but a single type of explanation may not work for everyone. Before considering how to achieve explainability, having a clear understanding of the explanations' end users is crucial. To effectively communicate the results to users and network operators, it is necessary to establish a suitable format that ensures usability. This highlights the **need for the development of interfaces that facilitate the presentation and interpretation of the explainability outcomes**. These interfaces play a vital role in conveying the obtained insights in a manner that can be readily comprehended and utilized by the intended recipients.

Equally important, there is an urgent **need for shared (and possibly rigorous) metrics to evaluate the explanations obtained**. Different techniques may return different results, and it is difficult to know which technique is to be preferred. Furthermore, having shared metrics allows for a standardized framework to assess the quality and effectiveness of explanations. By employing rigorous metrics, we can ensure that the evaluation process is objective and unbiased. This not only enhances transparency but also enables meaningful comparisons between different XAI techniques.

Unfortunately, NTA literature is still in its early stages in this respect, as highlighted by the scattered nature of relevant literature (Sec. V-A). Current evaluations are often not well-suited for evaluating the complex and multifaceted nature of explanations. While some metrics used in NTA focus on properties like Correctness and Continuity, these represent only two of twelve key properties of explainability [4]. On the other hand, for each of the twelve properties different metrics and techniques are available, and several of them require non-trivial computing (and thus a cost-time tradeoff): an exhaustive assessment of explanation quality would be impractical. Thus, there is a pressing need for a consensus on essential properties that cover the content, presentation, and user dimensions, providing a practical yet comprehensive framework. Tab. V marks a significant step toward a unified approach to explanation quality metrics in the NTA context, but achieving this goal will require collaboration among researchers and practitioners.

Linked to this topic, the sharing of datasets including annotated ground truth for explanations is crucial for evaluating new methods and selecting the most effective one. Networking poses even greater challenges compared to more established fields like computer vision, as even just assessing the plausibility of results—although straightforward for an image—becomes more complex in the case of network traffic. Consider, for instance, a model with payload as input that should be interpreted (even more complex if encrypted). Additional privacy-related issues are present regarding traffic trace sharing.

E. Absence of XAI-driven Foundation Models for Diverse NTA Tasks

At the time of writing, there is a recent and growing effort in developing foundation models that can suit different NTA tasks, but **none of them is built upon XAI guidelines**. Indeed, XAI can also assist network administrators in discovering the (otherwise hidden) security threats and loopholes in an interpretable way [20]. Accordingly, this will make such models easy to be tuned for different NTA tasks, but also *resilient* to different types of attacks specifically targeting the workflow of AI algorithms, such as evasion and privacy-leaking (e.g., membership inference). For instance, if the model behavior is changed due to attacks, XAI itself can be used as a potential detection mechanism against the attacks, even if the changes are subtle and not visible, merely observing model predictions [243]. Analyzing a model's resilience against attacks is essential in AI before integrating it into a real-world application in future networks. In this regard, a **lack of metrics for resilience evaluation** in the literature can be also observed.

Acknowledgments

This work is partially supported by the Italian Research Program "PON *Ricerca e Innovazione* 2014-2020 (PON R&I) REACT-EU – Asse IV – Azione IV.4", and by the "PNRR ICSC National Research Centre for High Performance Computing, Big Data and Quantum Computing (CN00000013)", under the NRRP MUR program funded by the NextGenerationEU. Also, this work is partially carried out within the "xInternet" Project supported by the MUR PRIN 2022 program (D.D.104—02/02/2022) funded by the NextGenerationEU. D. Ciuonzo acknowledges the support of the Marie Skłodowska-Curie Actions under the European Union's Horizon Europe research and innovation program for the Industrial Doctoral Network on Digital Finance (acronym: DIGITAL, project no. 101119635). Finally, this work is partially supported by the European Union under the Italian National Recovery and Resilience Plan (NRRP) of NextGenerationEU, partnership on "Telecommunications of the Future" (PE00000001 – program "RESTART").

References

[1] D. Rossi and L. Zhang, "Landing AI on networks: An equipment vendor viewpoint on autonomous driving networks," *IEEE Transactions on Network and Service Management*, vol. 19, no. 3, pp. 3670–3684, 2022.

[2] G. Aceto, D. Ciuonzo, A. Montieri, V. Persico, and A. Pescape, "AI-powered Internet Traffic Classification: Past, Present, and Future," *IEEE Communications Magazine*, pp. 1–7, 2023.

[3] M. Flora, C. Potvin, A. McGovern, and S. Handler, "Comparing Explanation Methods for Traditional Machine Learning Models Part 1: An Overview of Current Methods and Quantifying Their Disagreement," *arXiv preprint arXiv:2211.08943*, 2022.

[4] M. Nauta, J. Trienes, S. Pathak, E. Nguyen, M. Peters, Y. Schmitt, J. Schlötterer, M. van Keulen, and C. Seifert, "From anecdotal evidence to quantitative evaluation methods: A systematic review on evaluating explainable AI," *ACM Computing Surveys*, 2022.

[5] C. Wohlin, "Guidelines for snowballing in systematic literature studies and a replication in software engineering," in *Proc. International Conference on Evaluation and Assessment in Software Engineering (EASE)*, 2014, pp. 1–10.

[6] "Ethics guidelines for trustworthy AI," https://digital-strategy.ec.europa.eu/en/library/ethics-guidelines-trustworthy-ai, 2018–2019.

[7] Huawei, "AI Security White Paper," Huawei Technologies Co., Ltd, White Paper, Oct 2018. [Online]. Available: https://www.huawei.com/en/trust-center/resources/ai-security-white-paper

[8] Telefónica, "Telefónica's Approach to the Responsible Use of AI," Telefónica, S.A., Tech. Rep., 2018. [Online]. Available: https://www.telefonica.com/en/commitment/how-we-work/business-principles/

[9] D. Gunning and D. Aha, "DARPA's explainable artificial intelligence (XAI) program," *AI Magazine*, vol. 40, no. 2, pp. 44–58, 2019.

[10] A. Mujumdar, K. Čyras, S. Singh, and A. Vulgarakis, "Trustworthy AI: explainability, safety and verifiability," Ericsson, Tech. Rep., Dec 2020. [Online]. Available: https://www.ericsson.com/en/blog/2020/12/trustworthy-ai

[11] "EU AI Act," https://artificialintelligenceact.eu/the-act/, 2021–2024.

[12] NEC, "NEC AI Guide Book," NEC, White Paper, 2021. [Online]. Available: https://www.nec.com/en/global/solutions/ai/download/necaiguidebook/NEC_AI_Guide_Book_en.pdf

[13] "AI Innovation Principles," https://web.archive.org/web/20220523025005/https://www.juniper.net/us/en/company/ai-innovation-principles.html, first archived: 2022; Last accessed: 2024.

[14] A. Lee, "Responsible AI for telecom. The next step," Nokia, White Paper, 2023. [Online]. Available: https://onestore.nokia.com/asset/f/212898

[15] "Executive Order on the Safe, Secure, and Trustworthy Development and Use of Artificial Intelligence," https://www.whitehouse.gov/briefing-room/presidential-actions/2023/10/30/executive-order-on-the-safe-secure-and-trustworthy-development-and-use-of-artificial-intelligence/, 2023.

[16] Cisco, "Cisco Principles for Responsible Artificial Intelligence," Cisco, White Paper, 2024. [Online]. Available: https://www.cisco.com/c/dam/en_us/about/doing_business/trust-center/docs/cisco-responsible-artificial-intelligence-principles.pdf

[17] S. Hariharan, A. Velicheti, A. Anagha, C. Thomas, and N. Balakrishnan, "Explainable Artificial Intelligence in Cybersecurity: A Brief Review," in *Proc. IEEE International Conference on Security and Privacy (ISEA-ISAP)*, 2021, pp. 1–12.

[18] S. K. Jagatheesaperumal, Q.-V. Pham, R. Ruby, Z. Yang, C. Xu, and Z. Zhang, "Explainable AI over the Internet of Things (IoT): Overview, state-of-the-art and future directions," *IEEE Open Journal of the Communications Society*, 2022.

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

[20] T. Zhang, H. Qiu, M. Mellia, Y. Li, H. Li, and K. Xu, "Interpreting AI for networking: Where we are and where we are going," *IEEE Communications Magazine*, vol. 60, no. 2, pp. 25–31, 2022.

[21] N. Capuano, G. Fenza, V. Loia, and C. Stanzione, "Explainable artificial intelligence in cybersecurity: A survey," *IEEE Access*, vol. 10, pp. 93 575–93 600, 2022.

[22] S. Neupane, J. Ables, W. Anderson, S. Mittal, S. Rahimi, I. Banicescu, and M. Seale, "Explainable Intrusion Detection Systems (x-IDS): A survey of current methods, challenges, and opportunities," *IEEE Access*, vol. 10, pp. 112 392–112 415, 2022.

[23] D. Warmsley, A. Waagen, J. Xu, Z. Liu, and H. Tong, "A Survey of Explainable Graph Neural Networks for Cyber Malware Analysis," in *Proc. IEEE International Conference on Big Data (Big Data)*, 2022, pp. 2932–2939.

[24] C. Fiandrino, G. Attanasio, M. Fiore, and J. Widmer, "Toward native explainable and robust AI in 6G networks: Current state, challenges and road ahead," *Computer Communications*, vol. 193, pp. 47–52, 2022.

[25] G. Rjoub, J. Bentahar, O. A. Wahab, R. Mizouni, A. Song, R. Cohen, H. Otrok, and A. Mourad, "A Survey on Explainable Artificial Intelligence for Cybersecurity," *IEEE Transactions on Network and Service Management*, 2023.

[26] I. Kök, F. Y. Okay, Ö. Muyanlı, and S. Özdemir, "Explainable Artificial Intelligence (XAI) for Internet of Things," *IEEE Internet of Things Journal*, vol. 10, no. 16, pp. 14 764–14 779, 2023.

[27] N. Moustafa, N. Koroniotis, M. Keshk, A. Y. Zomaya, and Z. Tari, "Explainable Intrusion Detection for Cyber Defences in the Internet of Things: Opportunities and Solutions," *IEEE Communications Surveys & Tutorials*, vol. 25, no. 3, pp. 1775–1807, 2023.

[28] S. Wang, M. A. Qureshi, L. Miralles-Pechuán, T. Huynh-The, T. R. Gadekallu, and M. Liyanage, "Explainable ai for 6g use cases: Technical aspects and research challenges," *IEEE Open Journal of the Communications Society*, 2024.

[29] T. Senevirathna, Z. Salazar, V. H. La, S. Marchal, B. Siniarski, M. Liyanage, and S. Wang, "A survey on XAI for beyond 5G security: technical aspects, use cases, challenges and research directions," *IEEE Communications Surveys & Tutorials*, 2024.

[30] M. Danilevsky, K. Qian, R. Aharonov, Y. Katsis, B. Kawas, and P. Sen, "A Survey of the State of Explainable AI for Natural Language Processing," in *Proc. Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*. Association for Computational Linguistics, 2020, pp. 447–459. [Online]. Available: https://aclanthology.org/2020.aacl-main.46

[31] E. Tjoa and C. Guan, "A survey on explainable artificial intelligence (XAI): Toward medical XAI," *IEEE Transaction on Neural Networks Learning Systems*, vol. 32, no. 11, pp. 4793–4813, 2020.

[32] R. Machlev, L. Heistrene, M. Perl, K. Levy, J. Belikov, S. Mannor, and Y. Levron, "Explainable Artificial Intelligence (XAI) techniques for energy and power systems: Review, challenges and opportunities," *Energy and AI*, vol. 9, p. 100169, 2022.

[33] M. Abbasi, A. Shahraki, and A. Taherkordi, "Deep learning for network traffic monitoring and analysis (NTMA): A survey," *Computer Communications*, vol. 170, pp. 19–41, 2021.

[34] E. Papadogiannaki and S. Ioannidis, "A survey on encrypted network traffic analysis applications, techniques, and countermeasures," *ACM Computing Surveys (CSUR)*, vol. 54, no. 6, pp. 1–35, 2021.

[35] Z. C. Lipton, "The mythos of model interpretability: In machine

learning, the concept of interpretability is both important and slippery." *ACM Queue*, vol. 16, no. 3, pp. 31–57, 2018.

[36] R. Saleem, B. Yuan, F. Kurugollu, A. Anjum, and L. Liu, "Explaining deep neural networks: A survey on the global interpretation methods," *Neurocomputing*, vol. 513, pp. 165–180, 2022.

[37] M. Setzu, R. Guidotti, A. Monreale, F. Turini, D. Pedreschi, and F. Giannotti, "Glocalx-from local to global explanations of black box AI models," *Artificial Intelligence*, vol. 294, p. 103457, 2021.

[38] 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, 2021.

[39] E. Ferrara, "Fairness And Bias in Artificial Intelligence: A Brief Survey of Sources, Impacts, And Mitigation Strategies," *Sci*, vol. 6, no. 1, p. 3, 2023.

[40] D. V. Carvalho, E. M. Pereira, and J. S. Cardoso, "Machine learning interpretability: A survey on methods and metrics," *Electronics*, vol. 8, no. 8, p. 832, 2019.

[41] S. Wiegreffe and Y. Pinter, "Attention is not not explanation," in *Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, Nov. 2019, pp. 11–20.

[42] S. Dash, O. Gunluk, and D. Wei, "Boolean decision rules via column generation," in *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, vol. 31. Curran Associates, Inc., 2018.

[43] J. H. Friedman and B. E. Popescu, "Predictive Learning via Rule Ensembles," *JSTOR The Annals of Applied Statistics*, pp. 916–954, 2008.

[44] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proc. European Conference on Computer Vision (ECCV)*, 2014, pp. 818–833.

[45] L. Van der Maaten and G. Hinton, "Visualizing data using t-SNE." *Journal of Machine Learning Research*, vol. 9, no. 11, 2008.

[46] M. Robnik-Šikonja and M. Bohanec, "Perturbation-based explanations of prediction models," *Human and Machine Learning: Visible, Explainable, Trustworthy and Transparent*, pp. 159–175, 2018.

[47] A. Fisher, C. Rudin, and F. Dominici, "All Models are Wrong, but Many are Useful: Learning a Variable's Importance by Studying an Entire Class of Prediction Models Simultaneously." *Journal of Machine Learning Research*, vol. 20, no. 177, pp. 1–81, 2019.

[48] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, vol. 30. Curran Associates, Inc., 2017.

[49] M. T. Ribeiro, S. Singh, and C. Guestrin, "" Why should I trust you?" Explaining the predictions of any classifier," in *Proc. ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2016, pp. 1135–1144.

[50] W. Guo, D. Mu, J. Xu, P. Su, G. Wang, and X. Xing, "Lemna: Explaining deep learning based security applications," in *Proc. ACM SIGSAC Conference on Computer and Communications Security (CCS)*, 2018, pp. 364–379.

[51] M. Ancona, E. Ceolini, C. Öztireli, and M. Gross, "Gradient-based attribution methods," *Springer Explainable AI: Interpreting, explaining and visualizing deep learning*, pp. 169–191, 2019.

[52] S. Bach, A. Binder, G. Montavon, F. Klauschen, K.-R. Müller, and W. Samek, "On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation," *Public Library of Science One*, vol. 10, no. 7, p. e0130140, 2015.

[53] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *Proc. IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 618–626.

[54] M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic attribution for deep networks," in *Proc. PMLR International Conference on Machine Learning (ICML)*. PMLR, 2017, pp. 3319–3328.

[55] A. Shrikumar, P. Greenside, and A. Kundaje, "Learning important features through propagating activation differences," in *Proc. PMLR International Conference on Machine Learning (ICML)*, 2017, pp. 3145–3153.

[56] S. Wachter, B. Mittelstadt, and C. Russell, "Counterfactual explanations without opening the black box: Automated decisions and the GDPR," *HeinOnline Harv. JL & Tech.*, vol. 31, p. 841, 2017.

[57] A. Dhurandhar, P.-Y. Chen, R. Luss, C.-C. Tu, P. Ting, K. Shanmugam, and P. Das, "Explanations based on the missing: Towards contrastive explanations with pertinent negatives," in *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, vol. 31. Curran Associates, Inc., 2018.

[58] K. S. Gurumoorthy, A. Dhurandhar, G. Cecchi, and C. Aggarwal, "Efficient data representation by selecting prototypes with importance weights," in *Proc. IEEE International Conference on Data Mining (ICDM)*, 2019, pp. 260–269.

[59] L. Yang, W. Guo, Q. Hao, A. Ciptadi, A. Ahmadzadeh, X. Xing, and G. Wang, "CADE: Detecting and explaining concept drift samples for security applications," in *Proc. USENIX Security Symposium*, 2021, pp. 2327–2344.

[60] N. Liu, D. Shin, and X. Hu, "Contextual outlier interpretation," in *Proc. International Joint Conference on Artificial Intelligence (IJCAI)*, 2018, pp. 2461–2467.

[61] M. Kull, M. Perello Nieto, M. Kängsepp, T. Silva Filho, H. Song, and P. Flach, "Beyond temperature scaling: Obtaining well-calibrated multi-class probabilities with dirichlet calibration," in *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, vol. 32. Curran Associates, Inc., 2019.

[62] A. Niculescu-Mizil and R. Caruana, "Predicting good probabilities with supervised learning," in *Proc. 22nd International Conference on Machine learning (ICML)*, 2005, pp. 625–632.

[63] J. Mukhoti, V. Kulharia, A. Sanyal, S. Golodetz, P. H. Torr, and P. K. Dokania, "Calibrating deep neural networks using focal loss," in *Proc. 34th Conference on Neural Information Processing Systems (NeurIPS)*, 2020.

[64] R. Müller, S. Kornblith, and G. Hinton, "When does label smoothing help?" in *Proc 32th Conference on Neural Information Processing Systems (NeurIPS)*, 2019.

[65] H. D. Kabir, A. Khosravi, M. A. Hosen, and S. Nahavandi, "Neural network-based uncertainty quantification: A survey of methodologies and applications," *IEEE access*, vol. 6, pp. 36 218–36 234, 2018.

[66] A. N. Angelopoulos, S. Bates *et al.*, "Conformal prediction: A gentle introduction," *Foundations and Trends® in Machine Learning*, vol. 16, no. 4, pp. 494–591, 2023.

[67] S. Rezaei, B. Kroencke, and X. Liu, "Large-scale mobile app identification using deep learning," *IEEE Access*, vol. 8, pp. 348–362, 2019.

[68] C. Beliard, A. Finamore, and D. Rossi, "Opening the deep pandora box: Explainable traffic classification," in *Proc. IEEE Conference Computer Communications Workshops (INFOCOM WKSHPS)*, 2020, pp. 1292–1293.

[69] X. Wang, S. Chen, and J. Su, "Real network traffic collection and deep learning for mobile app identification," *Hindawi Limited Wireless Communications and Mobile Computing*, vol. 2020, pp. 1–14, 2020.

[70] G. Aceto, D. Ciuonzo, A. Montieri, and A. Pescapé, "Toward effective mobile encrypted traffic classification through deep learning," *Neurocomputing*, vol. 409, pp. 306–315, 2020.

[71] A. Nascita, A. Montieri, G. Aceto, D. Ciuonzo, V. Persico, and A. Pescapé, "XAI meets mobile traffic classification: Understanding and improving multimodal deep learning architectures," *IEEE Transactions on Network and Service Management*, vol. 18, no. 4, pp. 4225–4246, 2021.

[72] Z. Li, Y. Liu, C. Zhang, W. Shan, H. Zhang, and X. Zhu, "Trustworthy Deep Learning for Encrypted Traffic Classification," *Research Square preprint*, 2022.

[73] L. Garcia, G. Bartlett, S. Ravi, H. Ibrahim, W. Hardaker, and E. Kline, "Explaining Deep Learning Models for Per-packet Encrypted Network Traffic Classification," in *Proc. IEEE International Symposium on Measurements & Networking (M&N)*, 2022, pp. 1–6.

[74] I. Guarino, G. Aceto, D. Ciuonzo, A. Montieri, V. Persico, and A. Pescapè, "Contextual counters and multimodal Deep Learning for activity-level traffic classification of mobile communication apps during COVID-19 pandemic," *Computer Networks*, vol. 219, p. 109452, 2022.

[75] Z. Song, Z. Zhao, F. Zhang, G. Xiong, G. Cheng, X. Zhao, S. Guo, and B. Chen, "I ^{2} RNN: An Incremental and Interpretable Recurrent Neural Network for Encrypted Traffic Classification," *IEEE Transactions on Dependable and Secure Computing*, pp. 1–14, 2023.

[76] P. Luo, J. Chu, and G. Yang, "IP packet-level encrypted traffic classification using machine learning with a light weight feature engineering method," *Journal of Information Security and Applications*, vol. 75, p. 103519, 2023.

[77] A. Nascita, A. Montieri, G. Aceto, D. Ciuonzo, V. Persico, and A. Pescapé, "Improving Performance, Reliability, and Feasibility in Multimodal Multitask Traffic Classification with XAI," *IEEE Transactions on Network and Service Management*, 2023.

[78] S. Jorgensen, J. Holodnak, J. Dempsey, K. de Souza, A. Raghunath, V. Rivet, N. DeMoes, A. Alejos, and A. Wollaber, "Extensible machine learning for encrypted network traffic application labeling via uncertainty quantification," *IEEE Transactions on Artificial Intelligence*, 2023.

[79] K. Fauvel, F. Chen, and D. Rossi, "A Lightweight, Efficient and Explainable-by-Design Convolutional Neural Network for Internet Traffic Classification," in *Proc. ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. Association for Computing Machinery, 2023, p. 4013–4023.

[80] C.-Y. Shin, J.-T. Park, U.-J. Baek, and M.-S. Kim, "A Feasible and Explainable Network Traffic Classifier Utilizing DistilBERT," *IEEE Access*, 2023.

[81] S. Chowdhury, B. Liang, and A. Tizghadam, "Explaining class-of-service oriented network traffic classification with superfeatures," in *Proc. ACM CoNEXT Workshop on Big DAta, Machine Learning and Artificial Intelligence for Data Communication Networks (Big-DAMA)*, 2019, pp. 29–34.

[82] M. He, L. Jin, and M. Song, "Interpretability Framework of Network Security Traffic Classification Based on Machine Learning," in *Proc. International Conference on Artificial Intelligence and Security (ICAIS)*, 2021, pp. 305–320.

[83] J. Luxemburk and T. Čejka, "Fine-grained TLS services classification with reject option," *Computer Networks*, vol. 220, p. 109467, 2023.

[84] J. Luxemburk, K. Hynek, and T. Čejka, "Encrypted traffic classification: the QUIC case," in *Proc. IEEE Network Traffic Measurement and Analysis Conference (TMA)*, 2023, pp. 1–10.

[85] D. R. Arikkat, P. Vinod, K. Rafidha Rehiman, R. A. Rasheed, and M. Conti, "XAITrafficIntell: Interpretable Cyber Threat Intelligence for Darknet Traffic Analysis," *Journal of Network and Systems Management*, vol. 32, no. 4, p. 88, 2024.

[86] E. Luis-Bisbé, V. Morales-Gómez, D. Perdices, and J. E. López de Vergara, "No pictures, please: Using explainable artificial intelligence to demystify cnns for encrypted network packet classification," *Applied Sciences*, vol. 14, no. 13, p. 5466, 2024.

[87] A. S. Jacobs, R. Beltiukov, W. Willinger, R. A. Ferreira, A. Gupta, and L. Z. Granville, "AI/ML for Network Security: The Emperor Has No Clothes," in *Proc. ACM SIGSAC Conference on Computer and Communications Security (CCS)*. Association for Computing Machinery, 2022, p. 1537–1551.

[88] T. Dahanayaka, G. Jourjon, and S. Seneviratne, "Dissecting traffic fingerprinting CNNs with filter activations," *Computer Networks*, vol. 206, p. 108770, 2022.

[89] Y. Wang, X. Yun, Y. Zhang, C. Zhao, and X. Liu, "A multi-scale feature attention approach to network traffic classification and its model explanation," *IEEE Transactions on Network and Service Management*, vol. 19, no. 2, pp. 875–889, 2022.

[90] J. Piet, D. Nwoji, and V. Paxson, "GGFAST: Automating Generation of Flexible Network Traffic Classifiers," in *Proc. ACM SIGCOMM 2023 Conference*, 2023, pp. 850–866.

[91] M. Jafari Siavoshani, A. Khajehpour, A. Z. Bideh, A. Gatmiri, and A. Taheri, "Machine learning interpretability meets TLS fingerprinting," *Springer Soft Computing*, vol. 27, no. 11, pp. 7191–7208, 2023.

[92] K. Amarasinghe, K. Kenney, and M. Manic, "Toward explainable deep neural network based anomaly detection," in *Proc. IEEE International Conference on Human System Interaction (HSI)*, 2018, pp. 311–317.

[93] D. L. Marino, C. S. Wickramasinghe, and M. Manic, "An adversarial approach for explainable AI in intrusion detection systems," in *Proc. IEEE Annual Conference of the IEEE Industrial Electronics Society (IECON)*, 2018, pp. 3237–3243.

[94] M. Szczepański, M. Choraś, M. Pawlicki, and R. Kozik, "Achieving explainability of intrusion detection system by hybrid oracle-explainer approach," in *Proc. IEEE International Joint Conference on neural networks (IJCNN)*, 2020, pp. 1–8.

[95] C. Tang, N. Luktarhan, and Y. Zhao, "SAAE-DNN: Deep learning method on intrusion detection," *Symmetry*, vol. 12, no. 10, p. 1695, 2020.

[96] D. Han, Z. Wang, W. Chen, Y. Zhong, S. Wang, H. Zhang, J. Yang, X. Shi, and X. Yin, "Deepaid: Interpreting and improving deep learning-based anomaly detection in security applications," in *Proc. ACM SIGSAC Conference on Computer and Communications Security (CCS)*, 2021, pp. 3197–3217.

[97] W. Xu, Y. Fan, and C. Li, "I2DS: interpretable intrusion detection system using autoencoder and additive tree," *Hindawi Limited Security and Communication Networks*, vol. 2021, pp. 1–9, 2021.

[98] S. Mane and D. Rao, "Explaining network intrusion detection system using explainable AI framework," *arXiv preprint arXiv:2103.07110*, 2021.

[99] Ł. Wawrowski, M. Michalak, A. Białas, R. Kurianowicz, M. Sikora, M. Uchroński, and A. Kajzer, "Detecting anomalies and attacks in network traffic monitoring with classification methods and XAI-based explainability," *Procedia Computer Science*, vol. 192, pp. 2259–2268, 2021.

[100] B. Mahbooba, M. Timilsina, R. Sahal, and M. Serrano, "Explainable artificial intelligence (XAI) to enhance trust management in intrusion detection systems using decision tree model," *Hindawi Limited Complexity*, vol. 2021, pp. 1–11, 2021.

[101] A. Yan, Z. Chen, H. Zhang, L. Peng, Q. Yan, M. U. Hassan, C. Zhao, and B. Yang, "Effective detection of mobile malware behavior based on explainable deep neural network," *Neurocomputing*, vol. 453, pp. 482–492, 2021.

[102] N. Burkart, M. Franz, and M. F. Huber, "Explanation framework for intrusion detection," in *Proc. Machine Learning for Cyber Physical Systems: Selected papers from the International Conference ML4CPS 2020*, 2021, pp. 83–91.

[103] Y. Wang, P. Wang, Z. Wang, and M. Cao, "An Explainable Intrusion Detection System," in *2021 IEEE 23rd Int Conf on High Performance Computing & Communications; 7th Int Conf on Data Science & Systems; 19th Int Conf on Smart City; 7th Int Conf on Dependability in Sensor, Cloud & Big Data Systems & Application (HPCC/DSS/SmartCity/DependSys)*, 2021, pp. 1657–1662.

[104] T.-T.-H. Le, H. Kim, H. Kang, and H. Kim, "Classification and explanation for intrusion detection system based on ensemble trees and SHAP method," *Sensors*, vol. 22, no. 3, p. 1154, 2022.

[105] Z. Abou El Houda, B. Brik, and L. Khoukhi, ""Why should I trust your IDS?": An explainable deep learning framework for intrusion detection systems in Internet of Things networks," *IEEE Open Journal of the Communications Society*, vol. 3, pp. 1164–1176, 2022.

[106] J. Ables, T. Kirby, W. Anderson, S. Mittal, S. Rahimi, I. Banicescu, and M. Seale, "Creating an explainable intrusion detection system using self organizing maps," in *Proc. IEEE Symposium Series on Computational Intelligence (SSCI)*. IEEE, 2022, pp. 404–412.

[107] 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, 2022.

[108] Z. Abou El Houda, B. Brik, and S.-M. Senouci, "A novel IoT-based explainable deep learning framework for intrusion detection systems," *IEEE Internet of Things Magazine*, vol. 5, no. 2, pp. 20–23, 2022.

[109] 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, 2022.

[110] M. N. Yilmaz and B. Bardak, "An Explainable Anomaly Detection Benchmark of Gradient Boosting Algorithms for Network Intrusion Detection Systems," in *Proc. IEEE Innovations in Intelligent Systems and Applications Conference (ASYU)*. IEEE, 2022, pp. 1–6.

[111] F. P. Caforio, G. Andresini, G. Vessio, A. Appice, and D. Malerba, "Leveraging grad-CAM to improve the accuracy of network intrusion detection systems," in *Proc. International Conference on Discovery Science*. Springer, 2021, pp. 385–400.

[112] S. Layeghy and M. Portmann, "On generalisability of machine learning-based network intrusion detection systems," *arXiv preprint arXiv:2205.04112*, 2022.

[113] S. R. Islam and W. Eberle, "Domain Knowledge-Aided Explainable Artificial Intelligence," in *Explainable Artificial Intelligence for Cyber Security: Next Generation Artificial Intelligence*. Springer, 2022, pp. 73–92.

[114] K. Fujita, T. Shibahara, D. Chiba, M. Akiyama, and M. Uchida, "Objection!: Identifying Misclassified Malicious Activities with XAI," in *Proc. IEEE International Conference on Communications (ICC)*, 2022, pp. 2065–2070.

[115] S. Hariharan, R. Rejimol Robinson, R. R. Prasad, C. Thomas, and N. Balakrishnan, "XAI for intrusion detection system: comparing explanations based on global and local scope," *Journal of Computer Virology and Hacking Techniques*, vol. 19, no. 2, pp. 217–239, 2023.

[116] M. Keshk, N. Koroniotis, N. Pham, N. Moustafa, B. Turnbull, and A. Y. Zomaya, "An explainable deep learning-enabled intrusion detection framework in IoT networks," *Information Sciences*, vol. 639, p. 119000, 2023.

[117] A. Rehman and A. Farrakh, "Explainable AI in Intrusion Detection Systems: Enhancing Transparency and Interpretability," *International Journal of Advanced Sciences and Computing*, vol. 2, no. 1, pp. 26–39, 2023.

[118] S. Layeghy and M. Portmann, "Explainable Cross-domain Evaluation of ML-based Network Intrusion Detection Systems," *Computers and Electrical Engineering*, vol. 108, p. 108692, 2023.

[119] Z. Jadidi and S. Pal, "Explainable Anomaly Detection in IoT Networks," in *Emerging Smart Technologies for Critical Infrastructure*. Springer, 2023, pp. 85–94.

[120] M. A. Mukhtar Bhatti, M. Awais, and A. Iqtidar, "Machine Learning based Intrusion Detection System for IoT Applications using Explainable AI," in *Proc. Asia Conference on Artificial Intelligence, Machine Learning and Robotics*, 2023, pp. 1–6.

[121] M. T. Masud, M. Keshk, N. Moustafa, and I. Linkov, "An Explainable Intrusion Discovery Framework for Assessing Cyber Resilience in the Internet of Things Networks," in *Proc. Future Technologies Conference (FTC)*, 2023, pp. 199–215.

[122] S. Wali and I. Khan, "Explainable AI and random forest based reliable intrusion detection system," *Authorea Preprints*, 2023.

[123] L. A. C. Ahakonye, C. I. Nwakanma, J. M. Lee, and D.-S. Kim, "Machine learning explainability for intrusion detection in the industrial internet of things," *IEEE Internet of Things Magazine*, vol. 7, no. 3, pp. 68–74, 2024.

[124] M. T. Islam, M. K. Syfullah, M. G. Rashed, and D. Das, "Bridging the gap: advancing the transparency and trustworthiness of network intrusion detection with explainable AI," *International Journal of Machine Learning and Cybernetics*, pp. 1–24, 2024.

[125] H. Li, F. Wei, and H. Hu, "Enabling dynamic network access control with anomaly-based IDS and SDN," in *Proc. of the ACM International Workshop on Security in Software Defined Networks & Network Function Virtualization (SDN-NFV Security)*, 2019, pp. 13–16.

[126] Q. P. Nguyen, K. W. Lim, D. M. Divakaran, K. H. Low, and M. C. Chan, "GEE: A gradient-based explainable variational autoencoder for network anomaly detection," in *Proc. IEEE Conference on Communications and Network Security (CNS)*. IEEE, 2019, pp. 91–99.

[127] J. H. Sejr, A. Zimek, and P. Schneider-Kamp, "Explainable detection of zero day web attacks," in *Proc. IEEE International Conference on Data Intelligence and Security (ICDIS)*, 2020, pp. 71–78.

[128] C. S. Kalutharage, X. Liu, and C. Chrysoulas, "Explainable AI and Deep Autoencoders Based Security Framework for IoT Network Attack Certainty," in *Proc. International Workshop on Attacks and Defenses for Internet-of-Things (ADIoT)*. Springer, 2022, pp. 41–50.

[129] K. Roshan and A. Zafar, "Using Kernel SHAP XAI method to optimize the network anomaly detection model," in *Proc. IEEE International Conference on Computing for Sustainable Global Development (INDIACom)*, 2022, pp. 74–80.

[130] C. S. Kalutharage, X. Liu, C. Chrysoulas, N. Pitropakis, and P. Papadopoulos, "Explainable AI-based DDOS attack identification method for IoT networks," *Computers*, vol. 12, no. 2, p. 32, 2023.

[131] T. Dias, N. Oliveira, N. Sousa, I. Praça, and O. Sousa, "A hybrid approach for an interpretable and explainable intrusion detection system," in *Proc. International Conference on Intelligent Systems Design and Applications (ISDA)*, 2021, pp. 1035–1045.

[132] D. L. Marino, C. S. Wickramasinghe, C. Rieger, and M. Manic, "Self-supervised and interpretable anomaly detection using network transformers," *arXiv preprint arXiv:2202.12997*, 2022.

[133] D. H. Jeong, J.-H. Cho, F. Chen, L. Kaplan, A. Jøsang, and S.-Y. Ji, "Interactive Web-Based Visual Analysis on Network Traffic Data," *Multidisciplinary Digital Publishing Institute Information*, vol. 14, no. 1, p. 16, 2023.

[134] C. Minh, K. Vermeulen, C. Lefebvre, P. Owezarski, and W. Ritchie, "An explainable-by-design ensemble learning system to detect unknown network attacks," in *Proc. International Conference on Network and Service Management (CNSM)*, 2023.

[135] A. Guerra-Manzanares, S. Nõmm, and H. Bahsi, "Towards the integration of a post-hoc interpretation step into the machine learning workflow for IoT botnet detection," in *Proc. IEEE International Conference On Machine Learning And Applications (ICMLA)*, 2019, pp. 1162–1169.

[136] S. Tabassum, N. Parvin, N. Hossain, A. Tasnim, R. Rahman, and M. I. Hossain, "IoT Network Attack Detection Using XAI and Reliability Analysis," in *Proc. IEEE International Conference on Computer and Information Technology (ICCIT)*, 2022, pp. 176–181.

[137] S. Sohail, Z. Fan, X. Gu, and F. Sabrina, "Multi-tiered Artificial Neural Networks model for intrusion detection in smart homes," *Intelligent Systems with Applications*, vol. 16, p. 200152, 2022.

[138] P. P. Kundu, T. Truong-Huu, L. Chen, L. Zhou, and S. G. Teo, "Detection and classification of botnet traffic using deep learning with model explanation," *IEEE Transactions on Dependable and Secure Computing*, 2022.

[139] A. Nascita, F. Cerasuolo, D. Di Monda, J. T. A. Garcia, A. Montieri, and A. Pescapè, "Machine and deep learning approaches for IoT attack classification," in *Proc. IEEE Conference Computer Communications Workshops (INFOCOM WKSHPS)*, 2022, pp. 1–6.

[140] B. Sharma, L. Sharma, and C. Lal, "Anomaly-Based DNN Model for Intrusion Detection in IoT and Model Explanation: Explainable Artificial Intelligence," in *Proc. International Conference on Computational Electronics for Wireless Communications (ICCWC)*. Springer, 2023, pp. 315–324.

[141] B. Sharma, L. Sharma, C. Lal, and S. Roy, "Explainable artificial intelligence for intrusion detection in IoT networks: A deep learning based approach," *Expert Systems with Applications*, p. 121751, 2023.

[142] M. Siganos, P. Radoglou-Grammatikis, I. Kotsiuba, E. Markakis, I. Moscholios, S. Goudos, and P. Sarigiannidis, "Explainable AI-based Intrusion Detection in the Internet of Things," in *Proc. International Conference on Availability, Reliability and Security (ARES)*, 2023, pp. 1–10.

[143] A. A. Rida, R. Amhaz, and P. Parrend, "Metrics for Evaluating Interface Explainability Models for Cyberattack Detection in IoT Data," in *Proc. Springer International Conference on Complex Computational Ecosystems*. Springer, 2023, pp. 180–192.

[144] R. Kalakoti, H. Bahsi, and S. Nõmm, "Explainable Federated Learning for Botnet Detection in IoT Networks," in *Proc. IEEE International Conference on Cyber Security and Resilience (CSR)*. IEEE, 2024, pp. 01–08.

[145] S. Wang, Z. Chen, L. Zhang, Q. Yan, B. Yang, L. Peng, and Z. Jia, "Trafficav: An effective and explainable detection of mobile malware behavior using network traffic," in *Proc. IEEE/ACM International Symposium on Quality of Service (IWQoS)*, 2016, pp. 1–6.

[146] K. Amarasinghe and M. Manic, "Improving user trust on deep neural networks based intrusion detection systems," in *Proc. Annual Conference of the IEEE Industrial Electronics Society (IECON)*. IEEE, 2018, pp. 3262–3268.

[147] C. Wu, A. Qian, X. Dong, and Y. Zhang, "Feature-oriented design of visual analytics system for interpretable deep learning based intrusion detection," in *Proc. IEEE International Symposium on Theoretical Aspects of Software Engineering (TASE)*, 2020, pp. 73–80.

[148] 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.

[149] Y. Feng and J. Li, "Toward explainable and adaptable detection and classification of distributed denial-of-service attacks," in *Proc. Deployable Machine Learning for Security Defense: First International Workshop, MLHat*, 2020, pp. 105–121.

[150] G. Andresini, A. Appice, F. P. Caforio, D. Malerba, and G. Vessio, "ROULETTE: A neural attention multi-output model for explainable network intrusion detection," *Expert Systems with Applications*, vol. 201, p. 117144, 2022.

[151] T.-L. Nguyen, X.-H. Nguyen, K.-H. Le *et al.*, "Enhancing Explainability of Machine Learning-based Intrusion Detection Systems," in *Proc. IEEE International Conference on Computing and Communication Technologies (RIVF)*, 2022, pp. 606–611.

[152] A. Šarčević, D. Pintar, M. Vranić, and A. Krajna, "Cybersecurity knowledge extraction using XAI," *Applied Sciences*, vol. 12, no. 17, p. 8669, 2022.

[153] P. Zhao, Z. Fan, Z. Cao, and X. Li, "Intrusion detection model using temporal convolutional network blend into attention mechanism," *IGI Global International Journal of Information Security and Privacy (IJISP)*, vol. 16, no. 1, pp. 1–20, 2022.

[154] A.-E. Malik, G. Andresini, A. Appice, and D. Malerba, "An XAI-based adversarial training approach for cyber-threat detection," in *2022 IEEE Intl Conf on Dependable, Autonomic and Secure Computing, Intl Conf on Pervasive Intelligence and Computing, Intl Conf on Cloud and Big Data Computing, Intl Conf on Cyber Science and Technology Congress (DASC/PiCom/CBDCom/CyberSciTech)*. IEEE, 2022, pp. 1–8.

[155] B. Hsupeng, K.-W. Lee, T.-E. Wei, and S.-H. Wang, "Explainable malware detection using predefined network flow," in *Porc. IEEE International Conference on Advanced Communication Technology (ICACT)*. IEEE, 2022, pp. 27–33.

[156] K. Sauka, G.-Y. Shin, D.-W. Kim, and M.-M. Han, "Adversarial robust and explainable network intrusion detection systems based on deep learning," *Applied Sciences*, vol. 12, no. 13, p. 6451, 2022.

[157] H. Zhang, Y. Chen, W. Liu, S. Zhuang, J. Sun, Y. Liu, and L. Geng, "A Shapley-based Lightweight Global Explainer for Network Intrusion Detection System," in *2022 IEEE 24th Int Conf on High Performance Computing & Communications; 8th Int Conf on Data Science & Systems; 20th Int Conf on Smart City; 8th Int Conf on Dependability in Sensor, Cloud & Big Data Systems & Application (HPCC/DSS/SmartCity/DependSys)*. IEEE, 2022, pp. 554–561.

[158] S. Pande and A. Khamparia, "Explainable Deep Neural Network Based Analysis On Intrusion Detection Systems," *Computer Science*, vol. 24, no. 1, 2023.

[159] D. Javeed, T. Gao, P. Kumar, and A. Jolfaei, "An Explainable and Re-

silient Intrusion Detection System for Industry 5.0," *IEEE Transactions on Consumer Electronics*, 2023.

[160] Y. Wang, L. Xu, W. Liu, R. Li, and J. Gu, "Network intrusion detection based on explainable artificial intelligence," *Wireless Personal Communications*, vol. 131, no. 2, pp. 1115–1130, 2023.

[161] W. Ge, Z. Cui, J. Wang, B. Tang, and X. Li, "Metacluster: a universal interpretable classification framework for cybersecurity," *IEEE Transactions on Information Forensics and Security*, 2024.

[162] G. Aceto, G. Bovenzi, D. Ciuonzo, A. Montieri, V. Persico, and A. Pescapé, "Characterization and prediction of mobile-app traffic using Markov modeling," *IEEE Transactions on Network and Service Management*, vol. 18, no. 1, pp. 907–925, 2021.

[163] A. Montieri, G. Bovenzi, G. Aceto, D. Ciuonzo, V. Persico, and A. Pescapè, "Packet-level prediction of mobile-app traffic using multitask deep learning," *Computer Networks*, vol. 200, p. 108529, 2021.

[164] I. Guarino, G. Aceto, D. Ciuonzo, A. Montieri, V. Persico, and A. Pescapé, "Fine-Grained Traffic Prediction of Communication-and-Collaboration Apps via Deep-Learning: a First Look at Explainability," in *Proc. IEEE Int. Conference on Communications (ICC)*, 2023.

[165] A. Morichetta, P. Casas, and M. Mellia, "EXPLAIN-IT: Towards explainable AI for unsupervised network traffic analysis," in *Proc. ACM CoNEXT Workshop on Big DAta, Machine Learning and Artificial Intelligence for Data Communication Networks (Big-DAMA)*, 2019, pp. 22–28.

[166] A. Dethise, M. Canini, and S. Kandula, "Cracking Open the Black Box: What Observations Can Tell Us About Reinforcement Learning Agents," in *Proc. ACM Workshop on Network Meets AI & ML (NetAI)*, 2019, p. 29–36.

[167] A. Terra, R. Inam, S. Baskaran, P. Batista, I. Burdick, and E. Fersman, "Explainability methods for identifying root-cause of SLA violation prediction in 5G network," in *Proc. IEEE Global Communications Conference (GLOBECOM)*, 2020, pp. 1–7.

[168] Z. Meng, M. Wang, J. Bai, M. Xu, H. Mao, and H. Hu, "Interpreting deep learning-based networking systems," in *Proc. Annual conference of the ACM Special Interest Group on Data Communication on the applications, technologies, architectures, and protocols for computer communication*, 2020, pp. 154–171.

[169] A. Dainotti, A. Pescapè, and K. C. Claffy, "Issues and future directions in Traffic Classification," *IEEE Network*, vol. 26, no. 1, pp. 35–40, 2012.

[170] Y. Bengio, I. Goodfellow, and A. Courville, *Deep learning*. MIT press Cambridge, MA, USA, 2017, vol. 1.

[171] D. Gaspar, P. Silva, and C. Silva, "Explainable AI for Intrusion Detection Systems: LIME and SHAP Applicability on Multi-Layer Perceptron," *IEEE Access*, 2024.

[172] D. Chou and M. Jiang, "A survey on data-driven network intrusion detection," *ACM Computing Surveys (CSUR)*, vol. 54, no. 9, pp. 1–36, 2021.

[173] G. O. Ferreira, C. Ravazzi, F. Dabbene, G. C. Calafiore, and M. Fiore, "Forecasting network traffic: A survey and tutorial with open-source comparative evaluation," *IEEE Access*, vol. 11, pp. 6018–6044, 2023.

[174] C. Fiandrino, E. Perez Gomez, P. Férnandez Pérez, H. Mohammadalizadeh, M. Fiore, J. Widmer *et al.*, "AICHRONOLENS: Advancing Explainability for Time Series AI Forecasting in Mobile Networks," in *IEEE International Conference on Computer Communications*, 2024.

[175] S. Ahn, J. Kim, S. Y. Park, and S. Cho, "Explaining deep learning-based traffic classification using a genetic algorithm," *IEEE Access*, vol. 9, pp. 4738–4751, 2020.

[176] Z. Hang, Y. Lu, Y. Wang, and Y. Xie, "Flow-MAE: Leveraging Masked AutoEncoder for Accurate, Efficient and Robust Malicious Traffic Classification," in *Proc. International Symposium on Research in Attacks, Intrusions and Defenses (RAID)*, 2023, pp. 297–314.

[177] C. Callegari, P. Ducange, M. Fazzolari, and M. Vecchio, "Explainable internet traffic classification," *Applied Sciences*, vol. 11, no. 10, p. 4697, 2021.

[178] A. Nascita, F. Cerasuolo, G. Aceto, D. Ciuonzo, V. Persico, and A. Pescapé, "Explainable Mobile Traffic Classification: the Case of Incremental Learning," in *Proc. Explainable and Safety Bounded, Fidelitous, Machine Learning for Networking*. Association for Computing Machinery, 2023, p. 25–31.

[179] S. Jain and B. C. Wallace, "Attention is not Explanation," in *Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, Jun. 2019, pp. 3543–3556.

[180] Y. Zheng, Z. Liu, X. You, Y. Xu, and J. Jiang, "Demystifying Deep Learning in Networking," in *Proc. ACM Asia-Pacific Workshop on Networking (APNet)*, 2018, p. 1–7.

[181] J. Snell, K. Swersky, and R. Zemel, "Prototypical networks for few-shot learning," in *Proc. Advances in Neural Information Processing Systems (NeurIPS)*, vol. 30, 2017.

[182] G. Bovenzi, D. Di Monda, A. Montieri, V. Persico, and A. Pescapé, "META MIMETIC: Few-Shot Classification of Mobile-App Encrypted Traffic via Multimodal Meta-Learning," in *Proc. International Teletraffic Congress (ITC)*, 2023.

[183] M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set," in *Proc. IEEE Symposium on Computational Intelligence for Security and Defense Applications (CISDA)*, 2009, pp. 1–6.

[184] G. Dray, C. Raissi, J. Brissaud, P. Poncelet, M. Roche, and M. Teisseire, "Web Analysis Traffic Challenge: Description and Results," in *Discovery Challenge ECML/PKDD*, 2007.

[185] S. Revathi and A. Malathi, "A detailed analysis on NSL-KDD dataset using various machine learning techniques for intrusion detection," *International Journal of Engineering Research & Technology (IJERT)*, vol. 2, no. 12, pp. 1848–1853, 2013.

[186] M. Dusi, F. Gringoli, and L. Salgarelli, "Quantifying the accuracy of the ground truth associated with Internet traffic traces," *Computer Networks*, vol. 55, no. 5, pp. 1158–1167, 2011.

[187] N. Moustafa and J. Slay, "UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)," in *Proc. IEEE Military Communications and Information Systems Conference (MilCIS)*, 2015, pp. 1–6.

[188] T. Bujlow, V. Carela-Español, and P. Barlet-Ros, "Independent comparison of popular DPI tools for traffic classification," *Computer Networks*, vol. 76, pp. 75–89, 2015.

[189] G. Maciá-Fernández, J. Camacho, R. Magán-Carrión, P. García-Teodoro, and R. Therón, "UGR '16: A new dataset for the evaluation of cyclostationarity-based network IDSs," *Computers & Security*, vol. 73, pp. 411–424, 2018.

[190] G. Draper-Gil, A. H. Lashkari, M. S. I. Mamun, and A. A. Ghorbani, "Characterization of encrypted and vpn traffic using time-related," in *Proc. International Conference on Information Systems Security and Privacy (ICISSP)*, 2016, pp. 407–414.

[191] A. H. Lashkari, G. D. Gil, M. S. I. Mamun, and A. A. Ghorbani, "Characterization of tor traffic using time based features," in *Proc. SciTePress International Conference on Information Systems Security and Privacy (ICISSP)*, vol. 2. SciTePress, 2017, pp. 253–262.

[192] W. Wang, M. Zhu, X. Zeng, X. Ye, and Y. Sheng, "Malware traffic classification using convolutional neural network for representation learning," in *Proc. IEEE International Conference on Information Networking (ICOIN)*, 2017, pp. 712–717.

[193] I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, "Toward generating a new intrusion detection dataset and intrusion traffic characterization." in *Proc. International Conference on Information Systems Security and Privacy (ICISSP)*, vol. 1, 2018, pp. 108–116.

[194] Y. Mirsky, T. Doitshman, Y. Elovici, and A. Shabtai, "Kitsune: an ensemble of autoencoders for online network intrusion detection," in *Network and Distributed Systems Security Symposium (NDSS)*, 2018.

[195] A. Sivanathan, H. H. Gharakheili, F. Loi, A. Radford, C. Wijenayake, A. Vishwanath, and V. Sivaraman, "Classifying IoT devices in smart environments using network traffic characteristics," *IEEE Transactions on Mobile Computing*, vol. 18, no. 8, pp. 1745–1759, 2018.

[196] J. Ren, D. J. Dubois, D. Choffnes, A. M. Mandalari, R. Kolcun, and H. Haddadi, "Information exposure from consumer IoT devices: A multidimensional, network-informed measurement approach," in *Proc. ACM Internet Measurement Conference*, 2019, pp. 267–279.

[197] J. Ren, D. Dubois, and D. Choffnes, "An international view of privacy risks for mobile apps," 2019. [Online]. Available: https://recon.meddle.mobi/papers/cross-market.pdf

[198] G. Aceto, D. Ciuonzo, A. Montieri, V. Persico, and A. Pescapé, "MIRAGE: Mobile-app traffic capture and ground-truth creation," in *Proc. IEEE Int. Conference on Computing Communication Security (ICCS)*. IEEE, 2019, pp. 1–8.

[199] I. Sharafaldin, A. H. Lashkari, S. Hakak, and A. A. Ghorbani, "Developing realistic distributed denial of service (DDoS) attack dataset and taxonomy," in *Proc. IEEE International Carnahan Conference on Security Technology (ICCST)*, 2019, pp. 1–8.

[200] N. Koroniotis, N. Moustafa, E. Sitnikova, and B. Turnbull, "Towards the development of realistic botnet dataset in the Internet of Things for network forensic analytics: Bot-IoT dataset," *Future Generation Computer Systems*, vol. 100, pp. 779–796, 2019.

[201] N. Moustafa, "A new distributed architecture for evaluating AI-based security systems at the edge: Network TON_IoT datasets," *Sustainable Cities and Society*, vol. 72, p. 102994, 2021.

[202] I. Ullah and Q. H. Mahmoud, "A scheme for generating a dataset for anomalous activity detection in IoT networks," in *Proc. Canadian Conference on Artificial Intelligence (Canadian AI)*. Springer, 2020, pp. 508–520.

[203] A. Habibi Lashkari, G. Kaur, and A. Rahali, "Didarknet: A contemporary approach to detect and characterize the darknet traffic using deep image learning," in *Proc. 10th International Conference on Communication and Network Security (ICCNS)*, 2020, pp. 1–13.

[204] M. Catillo, A. Del Vecchio, L. Ocone, A. Pecchia, and U. Villano, "USB-IDS-1: a public multilayer dataset of labeled network flows for IDS evaluation," in *Proc. Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshops (DSN-W)*. IEEE, 2021, pp. 1–6.

[205] I. Guarino, G. Aceto, D. Ciuonzo, A. Montieri, V. Persico, and A. Pescapé, "Classification of Communication and Collaboration Apps via Advanced Deep-Learning Approaches," in *Proc. IEEE International Workshop on Computer Aided Modeling and Design of Communication Links and Networks (CAMAD)*, 2021, pp. 1–6.

[206] C. Wang, A. Finamore, L. Yang, K. Fauvel, and D. Rossi, "AppClassNet: A commercial-grade dataset for application identification research," *ACM SIGCOMM Computer Communication Review*, vol. 52, no. 3, pp. 19–27, 2022.

[207] J. Luxemburk, K. Hynek, T. Čejka, A. Lukačovič, and P. Šiška, "CESNET-QUIC22: A large one-month QUIC network traffic dataset from backbone lines," *Data in Brief*, vol. 46, p. 108888, 2023.

[208] P. Radoglou-Grammatikis, K. Rompolos, P. Sarigiannidis, V. Argyriou, T. Lagkas, A. Sarigiannidis, S. Goudos, and S. Wan, "Modeling, detecting, and mitigating threats against industrial healthcare systems: a combined software defined networking and reinforcement learning approach," *IEEE Transactions on Industrial Informatics*, vol. 18, no. 3, pp. 2041–2052, 2021.

[209] A. Shiravi, H. Shiravi, M. Tavallaee, and A. A. Ghorbani, "Toward developing a systematic approach to generate benchmark datasets for intrusion detection," *Computers & Security*, vol. 31, no. 3, pp. 357–374, 2012.

[210] K. Shahbar and A. N. Zincir-Heywood, "Anon17: Network traffic dataset of anonymity services," Faculty of Computer Science, Dalhousie University, Tech. Rep., Mar. 2017.

[211] R. Wang, Z. Liu, Y. Cai, D. Tang, J. Yang, and Z. Yang, "Benchmark data for mobile app traffic research," in *Proc. EAI International Conference on Mobile and Ubiquitous Systems: Computing, Networking and Services (MobiQuitous)*, 2018, pp. 402–411.

[212] V. Tong, H. A. Tran, S. Souihi, and A. Mellouk, "A novel QUIC traffic classifier based on convolutional neural networks," in *Proc. IEEE Global Communications Conference (GLOBECOM)*, 2018, pp. 1–6.

[213] Y. Meidan, M. Bohadana, Y. Mathov, Y. Mirsky, A. Shabtai, D. Breitenbacher, and Y. Elovici, "N-baiot–network-based detection of IoT botnet attacks using deep autoencoders," *IEEE Pervasive Computing*, vol. 17, no. 3, pp. 12–22, 2018.

[214] H. Hindy, E. Bayne, M. Bures, R. Atkinson, C. Tachtatzis, and X. Bellekens, "Machine learning based IoT intrusion detection system: An MQTT case study (MQTT-IoT-IDS2020 dataset)," in *Springer International networking conference*, 2020, pp. 73–84.

[215] I. Akbari, M. A. Salahuddin, L. Ven, N. Limam, R. Boutaba, B. Mathieu, S. Moteau, and S. Tuffin, "A look behind the curtain: Traffic classification in an increasingly encrypted web," *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, vol. 5, no. 1, pp. 1–26, 2021.

[216] Y. Heng, V. Chandrasekhar, and J. G. Andrews, "UTMobileNetTraffic2021: A labeled public network traffic dataset," *IEEE Networking Letters*, vol. 3, no. 3, pp. 156–160, 2021.

[217] S. Dadkhah, H. Mahdikhani, P. K. Danso, A. Zohourian, K. A. Truong, and A. A. Ghorbani, "Towards the development of a realistic multidimensional IoT profiling dataset," in *Proc. IEEE Annual International Conference on Privacy, Security & Trust (PST)*, 2022, pp. 1–11.

[218] P. Jovanovic and D. Vuletic, "ETF IoT Botnet Dataset," 2021. [Online]. Available: https://data.mendeley.com/datasets/nbs66kvx6n/1

[219] M. A. Ferrag, O. Friha, D. Hamouda, L. Maglaras, and H. Janicke, "Edge-IIoTset: A New Comprehensive Realistic Cyber Security Dataset of IoT and IIoT Applications: Centralized and Federated Learning," 2022. [Online]. Available: https://dx.doi.org/10.21227/mbc1-1h68

[220] E. C. P. Neto, S. Dadkhah, R. Ferreira, A. Zohourian, R. Lu, and A. A. Ghorbani, "CICIoT2023: A real-time dataset and benchmark for large-scale attacks in IoT environment," *Sensors*, vol. 23, no. 13, p. 5941, 2023.

[221] V. Arya, R. K. Bellamy, P.-Y. Chen, A. Dhurandhar, M. Hind, S. C. Hoffman, S. Houde, Q. V. Liao, R. Luss, A. Mojsilović *et al.*, "One explanation does not fit all: A toolkit and taxonomy of AI explainability techniques," *arXiv preprint arXiv:1909.03012*, 2019.

[222] J. Klaise, A. Van Looveren, G. Vacanti, and A. Coca, "Alibi explain: Algorithms for explaining machine learning models," *Journal of Machine Learning Research*, vol. 22, no. 181, pp. 1–7, 2021.

[223] M. T. Ribeiro, S. Singh, and C. Guestrin, "Anchors: High-precision model-agnostic explanations," in *Proc. AAAI Conference on Artificial Intelligence*, 2018.

[224] N. Kokhlikyan, V. Miglani, M. Martin, E. Wang, B. Alsallakh, J. Reynolds, A. Melnikov, N. Kliushkina, C. Araya, S. Yan, and O. Reblitz-Richardson, "Captum: A unified and generic model interpretability library for PyTorch," 2020.

[225] H. Baniecki, W. Kretowicz, P. Piatyszek, J. Wisniewski, and P. Biecek, "Dalex: Responsible machine learning with interactive explainability and fairness in Python," *Journal of Machine Learning Research*, vol. 22, no. 1, pp. 9759–9765, 2021.

[226] R. K. Mothilal, A. Sharma, and C. Tan, "Explaining machine learning classifiers through diverse counterfactual explanations," in *Proc. Conference on Fairness, Accountability, and Transparency (FAccT)*, 2020, pp. 607–617.

[227] A. Sharma and E. Kiciman, "DoWhy: An End-to-End Library for Causal Inference," *arXiv preprint arXiv:2011.04216*, 2020.

[228] H. Nori, S. Jenkins, P. Koch, and R. Caruana, "Interpretml: A unified framework for machine learning interpretability," *arXiv preprint arXiv:1909.09223*, 2019.

[229] M. Alber, S. Lapuschkin, P. Seegerer, M. Hägele, K. T. Schütt, G. Montavon, W. Samek, K.-R. Müller, S. Dähne, and P.-J. Kindermans, "iNNvestigate neural networks!" *Journal of Machine Learning Research*, vol. 20, no. 93, pp. 1–8, 2019.

[230] W. Yang, H. Le, T. Laud, S. Savarese, and S. C. Hoi, "Omnixai: A library for explainable AI," *arXiv preprint arXiv:2206.01612*, 2022.

[231] S. A. Siddiqui, D. Mercier, M. Munir, A. Dengel, and S. Ahmed, "TSViz: Demystification of Deep Learning Models for Time-Series Analysis," *IEEE Access*, pp. 1–1, 2019.

[232] J. Wexler, M. Pushkarna, T. Bolukbasi, M. Wattenberg, F. Viégas, and J. Wilson, "The what-if tool: Interactive probing of machine learning models," *IEEE Transactions on Visualization and Computer Graphics*, vol. 26, no. 1, pp. 56–65, 2019.

[233] T. Fel, L. Hervier, D. Vigouroux, A. Poche, J. Plakoo, R. Cadene, M. Chalvidal, J. Colin, T. Boissin, L. Béthune *et al.*, "Xplique: A deep learning explainability toolbox," *arXiv preprint arXiv:2206.04394*, 2022.

[234] CalculatedContent, "Weight Watcher," https://github.com/CalculatedContent/WeightWatcher, 2018.

[235] R. K. E. Bellamy, K. Dey, M. Hind, S. C. Hoffman, S. Houde, K. Kannan, P. Lohia, J. Martino, S. Mehta, A. Mojsilovic, S. Nagar, K. N. Ramamurthy, J. Richards, D. Saha, P. Sattigeri, M. Singh, K. R. Varshney, and Y. Zhang, "AI Fairness 360: An Extensible Toolkit for Detecting, Understanding, and Mitigating Unwanted Algorithmic Bias," 2018. [Online]. Available: https://arxiv.org/abs/1810.01943

[236] P. Saleiro, B. Kuester, L. Hinkson, J. London, A. Stevens, A. Anisfeld, K. T. Rodolfa, and R. Ghani, "Aequitas: A bias and fairness audit toolkit," *arXiv preprint arXiv:1811.05577*, 2018.

[237] Fairlearn, "Fairlearn," https://github.com/fairlearn/fairlearn, 2024.

[238] K. Sokol, A. Hepburn, R. Poyiadzi, M. Clifford, R. Santos-Rodriguez, and P. Flach, "FAT Forensics: A Python Toolbox for Implementing and Deploying Fairness, Accountability and Transparency Algorithms in Predictive Systems," *Journal of Open Source Software*, vol. 5, no. 49, p. 1904, 2020. [Online]. Available: https://doi.org/10.21105/joss.01904

[239] M. Ali, "PyCaret: An open source, low-code machine learning library in Python," April 2020, pyCaret version 1.0. [Online]. Available: https://www.pycaret.org

[240] L. Longo, M. Brcic, F. Cabitza, J. Choi, R. Confalonieri, J. Del Ser, R. Guidotti, Y. Hayashi, F. Herrera, A. Holzinger *et al.*, "Explainable Artificial Intelligence (XAI) 2.0: A Manifesto of Open Challenges and Interdisciplinary Research Directions," *Information Fusion*, vol. 106, p. 102301, 2024.

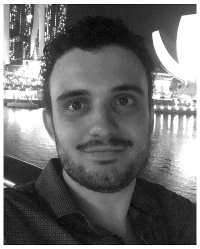
[241] C. Jean-Quartier, K. Bein, L. Hejny, E. Hofer, A. Holzinger, and F. Jeanquartier, "The Cost of Understanding–XAI Algorithms towards Sustainable ML in the View of Computational Cost," *Computation*, vol. 11, no. 5, p. 92, 2023.

[242] 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.

[243] C. Sandeepa, T. Senevirathna, B. Siniarski, M.-D. Nguyen, V.-H. La, S. Wang, and M. Liyanage, "From opacity to clarity: Leveraging XAI for robust network traffic classification," in *Proc. International*

Conference on Asia Pacific Advanced Network (APAN), 2023, pp. 125–138.

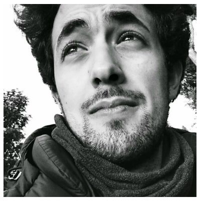
Alfredo Nascita is a PhD Candidate in Information Technology and Electrical Engineering at DIETI, University of Napoli Federico II. He received his M.S. Laurea Degree in Computer Engineering from the same University in March 2021. His research interests include traffic classification, machine and deep learning, and explainable artificial intelligence.



Antonio Montieri (M'18) is an Assistant Professor at DIETI of the University of Napoli Federico II. He has received his Ph.D. degree in Information Technology and Electrical Engineering in April 2020 from the same University. His work concerns network measurements, (encrypted and mobile) traffic classification, traffic modeling and prediction, and monitoring of cloud network performance. Antonio has co-authored more than 50 papers in international journals and conference proceedings.

Giuseppe Aceto is an Associate Professor at University of Napoli Federico II, where he received his PhD in Telecommunication Engineering. His research concerns network performance, traffic analysis, and censorship, both in traditional networks and SDN, and ICTs applied to health. He received the best paper award at IEEE ISCC 2010, and 2018 Best Journal Paper Award by IEEE CSIM.

Domenico Ciuonzo (S'11-M'14-SM'16) is Tenure-Track Professor at University of Napoli Federico II. He holds a Ph.D. from the University of Campania L. Vanvitelli. He is the recipient of two Paper awards (IEEE ICCCS 2019 and Elsevier ComNet 2020), the 2019 IEEE AESS Exceptional Service award, the 2020 IEEE SENSORS COUNCIL Early-Career Technical Achievement award and the 2021 IEEE AESS Early-Career Award. His research interests are data fusion, network analytics, IoT, and AI.



Valerio Persico is a Tenure-Track Professor at the University of Napoli Federico II, where he received the PhD in Computer and Automation Engineering in 2016. His work concerns network measurements, traffic analysis, cloud-network monitoring, and Internet path tracing. He has co-authored more than 70 papers within international journals and conference proceedings and is the recipient of several awards (including IEEE ISCC 2022, IEEE ICCCS 2019, IEEE CSIM 2018, and ACM CoNEXT 2013-Student Workshop).

Antonio Pescapé (SM'09) is a Full Professor of computer engineering at the University of Napoli Federico II. His work focuses on measurement, monitoring, and analysis of the Internet. He has co-authored more than 200 conference and journal papers, he is the recipient of a number of research awards. Also, he has served as an independent reviewer/evaluator of research projects/project proposals co-funded by a number of governments and agencies.