Anomaly Detection using Knowledge Graphs: A Survey for Network Management and Cybersecurity Application

Lionel Tailhardat, Yoan Chabot, Raphaël Troncy

Anomaly Detection using Knowledge Graphs: A Survey for Network Management and Cybersecurity Application

Lionel Tailhardat[1,2*], Yoan Chabot[1] and Raphael Troncy[2]

[1]Orange, France.
[2]EURECOM, France.

*Corresponding author(s). E-mail(s): lionel.tailhardat@orange.com;
Contributing authors: yoan.chabot@orange.com;
raphael.troncy@eurecom.fr;

Abstract

Incident management on telecom and computer networks, whether it is related to infrastructure or cybersecurity issues, requires the ability to simultaneously and quickly correlate and interpret a large number of heterogeneous technical information sources. Drawing on the understanding that knowledge representation and reasoning are inherently linked, this survey scrutinizes both aspects in tandem by delving into explicit knowledge representations of networks, and exploring their direct utilization or integration with artificial intelligence techniques for anomaly model learning and detection. More formally, we map these two aspects in order to address the question of how to define an anomaly model in a dynamic technical environment with various interdependencies, and what form this model should take to be shareable among practitioners (network designers and administrators, cybersecurity analysts, etc.) and directly usable in anomaly detection tools and decision support systems. Through our work, we demonstrate that while data heterogeneity and interrelatedness between data entities (distinct and persistent units of information) appear to be cornerstones for advancing the capabilities of Network Monitoring System (NMS) and Security Information and Event Management (SIEM) systems, several semantic models and algorithmic methods for anomaly detection share common properties that could help constructing a rich representation of networks and their ecosystem that can be used by one or a combination of several inference techniques.

Keywords: Network operations, Incident management, Semantic Web, Linked data, Reasoning, Statistical learning

MSC Classification: 68-02 , 68M15 , 68T30 , 68T37 , 68U35

1 Introduction

When managing large-scale IT and telco networks (broadband international backbones, corporate networks, Internet access networks), one is sooner or later involved into handling complex incident situations, such as general IT service disruption because of cascading failures or cyber-attacks. For incident management, technical support teams typically leverage information from decision support tools like Network Monitoring Systems (NMSs) or Security Information and Event Management (SIEM) systems. These tools often use an elementary representation of the network infrastructures and services. Basically, an IT network is a set of computers, routers, and other devices connected and configured to allow data processing and sharing. Similarly, an IT service is the usage of this processing and sharing capability for specific purposes, from the most trivial ones (entertainment, ticket booking, home automation) to more challenging ones (stock exchange, road lights, or nuclear plant management).

Although obvious at first glance, this level of description is not sufficient to scale up for maintaining high-standard quality of service on large-scale networks. This is due to the heterogeneity of the Information and Communications Technology (ICT) systems that compose them, making incident diagnosis and remediation a challenging task: to ensure network services function properly, supervision teams need to understand information from diverse and dynamic technical systems. For example, one can consider a service architecture that combines Virtual Machines (VM) distributed across data centers, which are interconnected through an IPoDWDM[1] network. To achieve efficiency, it is necessary to integrate and correlate data from various sources. This includes data from VM management tools, Optical Transport Network (OTN) layer management tools (which may be managed by a third-party operator), information about scheduled operations, and contact details for local servicing teams. Given the interdependencies between services and infrastructure and the inherent complexity of networks, the importance of a comprehensive and standardized knowledge representation of network assets and events therefore becomes evident for anyone wishing to develop a decision support system that can capture and analyze an incident context in its full complexity.

At the same time, one might be tempted to solve these complexity and operational efficiency challenges by adding artificial intelligence techniques to monitoring tools. This is already observed in various commercial and open source products for alarm grouping, alarm prioritization, alerting on trend breaks (e.g. sudden increase in network traffic), or alerting on risky user behaviors (e.g. unusually frequent authentication attempts from various sites). It is typically implemented through a business rules system and an overlay of correlation analysis. This approach is effective in that the business rules ensure a form of explainability for the generated alerts and recommendations thanks to their explicit and logical form. However, the operational burden

[1] Internet Protocol (IP) over Dense Wavelength-Division Multiplexing (DWDM).

remains significant because rule-based systems are complicated to maintain due to a great number of fine-grained rules and typically react to discrete stimuli, which hinders the generalization of rules for complex networks and often leads to missing the detection of anomalies (false negatives). Another approach is the use of probabilistic models derived from machine learning. It is also effective in that it allows for generalization to different types of stimuli but sacrifices explainability because of untractable model representations (e.g. the weight matrices of a deep neural network) and introduces an operational burden due to the need to qualify falsely generated alerts (false positives). Given that both approaches seem to have complementary advantages and disadvantages, the question arises of identifying principles that can be shared to meet the requirements of explainability and generalization, in line with the earlier mentioned need for standardized representation.

With this survey, we propose to tackle the challenges inherent in ICT systems operations by exploring the conditions for developing a network and IT monitoring system with advanced anomaly detection and reasoning capabilities through the lenses of knowledge representation and reasoning: we provide a comprehensive overview with perspectives about graph-based knowledge representations, existing anomaly detection methods, and how these two perspectives can match. Through this, we consider the possibility of improving operational efficiency in incident management situations and enhancing the design of complex network architectures by learning an explicit representation of the context of incidents (i.e. including information about system configuration and events that have occurred).

The remainder of this article is organized as follows: Section 2 presents a description of the application domain by providing background knowledge on *Network Design* and *Network Operations and Incident Management*. Section 3 defines challenges inherent in ICT systems operations that will guide the remainder of the survey, as well as a list of associated working hypotheses to help define a potential design project for next-generation Decision Support Systems (DSSs). Section 4 outlines the research methodology used for this survey. Section 5 examines DSSs in the Network administration and Operations (NetOps) and Cybersecurity Operations (SecOps) fields, and identifies limitations that need to be addressed based on their capabilities and the expectations of these fields. Section 6 examines semantic models for storing and managing technical and operational data required for NetOps and SecOps activities. Section 7 examines algorithmic methods for anomaly detection from the literature, focusing on application domains close to NetOps and SecOps. It provides an overall analysis of these methods, considering their principles, practicality within the incident management process, and the data structures involved. Finally, Section 8 concludes the survey by summarizing the technological and scientific challenges and suggesting future directions for knowledge graph-based monitoring systems. In Appendix A, we provide the detailed information that enabled the construction of the analyses in Section 7.

2 Background

In this section, we provide background knowledge related to this survey. We first focus on *network design* (Section 2.1) by defining what a network is and what characterizes its dynamics. We then address *network operations and incident management* (Section 2.2) by defining what anomalies are and how they are managed within a business process.

2.1 Network Design

What are networks?

An IT network is a set of computers, routers, and other devices connected and configured to allow data processing and sharing. Similarly, an IT service is the usage of this processing and sharing capability for specific purposes. The standardization efforts in the telecom and IT industry have led to standards and specifications for data transmission and processing system interfaces, such as the OSI model [1]. The OSI model allows for a description of network infrastructures and services along horizontal and vertical axes, representing the technical assets and protocol stack required for data transmission (Figure 1).

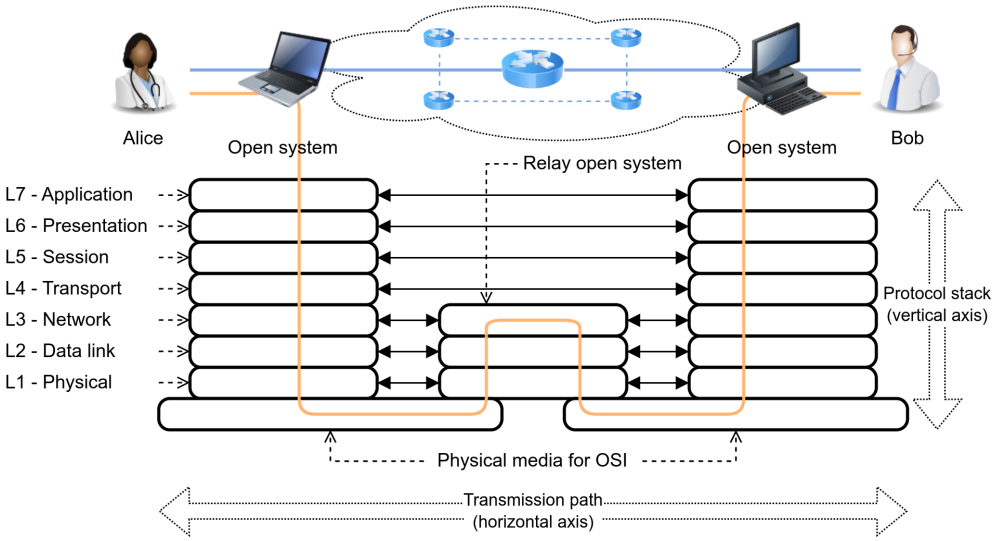

Fig. 1: A flat view of a network with a data flow, and its equivalent in the OSI model. Top: schematical view of an IT network. The central bubble represents a simplified network consisting of a mesh network of routers (i.e. network elements that determine and ensure the routing of data packets). The router in the center is an abstract representation of the redundancy group formed by the mesh of routers. Bottom: data path along the OSI model. The horizontal axis gives us a lecture of the technical assets (e.g. laptop, ethernet switch, router, firewall, etc.) involved in handling message data. The vertical axis provides insights into the protocol stack required for data takeover (i.e. encoding, routing, presentation, etc.). Based on the TISO2930-94/d11 diagram from [1].

According to this framework, routing a payload data unit from point A to B involves a horizontal data path formed by a set of network elements (Eq. 1, where

tp stands for transmission path, and ne for network element), such as when Alice is sending an "hello" message to Bob using an instant messaging service.[2]

$$tp_{A\to B} = A \to ne_1 \to \cdots ne_i \cdots \to ne_N \to B \tag{1}$$

The vertical perspective considers the protocol stacking within each network element, including line coding and error correction services of L1, encapsulation, link and medium access control of L2 microcode, L3 routing engine, etc., up to L7 if relevant. Combining both axes allows for stack-to-stack connections between protocol endpoints, such as from the L7 layer on Alice's terminal to the L7 layer on Bob's terminal in Figure 1.

Network dynamics.

The previous description corresponds to a scenario where data flows along a fixed transmission path. However, considerations for network dependability introduce the use of a mesh network architecture and failover mechanisms, which bring temporal reachability concerns. In terms of the horizontal axis, a temporarily unavailable network element is automatically replaced by another, ensuring the continuity of the data path from the end users' perspective (Eq. 2), where $t1$ and $t2$ are two different instants.

$$\begin{aligned} &\left(tp_{A\to B}(t1) = A \to ne_1^{t1} \to \cdots ne_i^{t1} \cdots \to ne_N^{t1} \to B\right) \\ \equiv &\left(tp_{A\to B}(t2) = A \to ne_1^{t2} \to \cdots ne_i^{t2} \cdots \to ne_N^{t2} \to B\right) \end{aligned} \tag{2}$$

This leads to a functional object that remains invariant over time, represented by a pseudo ordered set (Eq. 3), where $\{ne_{i,j}\}$ is a set of network elements that forms a redundancy group (i.e. assets within tp that are functionally equivalent).

$$tp^*_{A\to B} = A \to \cdots \{ne_{i,j}\} \cdots \to B \tag{3}$$

The dynamics of networks must also be viewed through the multiplexing and virtualization capabilities offered by the networks. These capabilities are primarily motivated by considerations of optimal resource allocation and rapid deployment in relation to the expectations for the network/service functions (e.g. forwarding, filtering, processing) and performance (e.g. throughput, number of simultaneous users, latency). Multiplexing can involve parallelizing links ($tp_{A\to B} = A \to \cdots \left\{tp_{a1\to b1} \parallel \cdots \parallel tp_{aN\to bN}\right\} \cdots \to B$) to increase throughput or path resilience. Multiplexing can also mean partitioning or stacking flows through recursive encapsulation ($\left\{tp_{a1\to b1} \parallel \cdots \parallel tp_{aN\to bN}\right\} \subset tp_{A\to B}$) to maximize the reuse of a given data transmission resource. Finally, virtualization enables temporary and mobile processing capacity based on processing resource sharing ($(ne_i \subset ne_j) \prec (ne_i \subset ne_k)$), like deploying a firewall based on user needs or a cache system based on the localization of users.

[2]Implementation details, such as bidirectional mechanisms for flow control or data link management (e.g. IP/TCP "SYN/SYN-ACK/ACK" sequence), are not discussed here for simplicity.

2.2 Network Operations and Incident Management

What is an "anomaly"?

The term "anomaly" commonly refers to a deviation from the normal state that requires action for recovery. In the context of NetOps, an anomaly is defined with respect to a supervision process[3] where **alarms** are a logical consequence of **errors** (i.e. a deviation of a system from normal operation [3]) caused by persistent **fault causes** (i.e. the physical or algorithmic cause of a malfunction [3]). These faults occur within the atomic functions of network devices [2]. The alarms should be reported to a Management and Control System, such as a Computerized Maintenance Management System or Software Defined Network controller. The logical sequence of these concepts can be summarized by: $Fault \succ Error \succ Alarm \succ AlarmReport$.

In the context of SecOps, an anomaly is defined based on a business policy: user and device activities that comply with the policy are considered legitimate, while others may be classified as **attack**[4]. Errors, alarms and other automated analysis reports from both the NetOps and SecOps domains serve as input for a behavioral analysis process that utilizes causal entities such as **threats** (i.e. entities that can adversely act on an unwanted asset [5]) and **vulnerabilities** (i.e. weaknesses of assets that can be exploited by threats [5]). This analysis helps detecting attacks and **incidents** (i.e. unwanted events resulting in the loss of confidentiality, integrity, and/or availability [5]).

Incident management and root cause analysis.

The concept of IT Service Management (ITSM) emerged in the 70s-80s as organizations recognized the importance of Information Technology for their operational efficiency. Standards and best practices, such as ISO/IEC 20000 [6], ITIL [7], and FitSM [8], were developed to provide guidance to Information Technology organizations in aligning their ITSM processes with business needs and international best practices. These standards emphasize the establishment of a continuous quality improvement loop, which relies on the observability of ICT systems and the accumulation of knowledge, such as the causes of incidents and the corrective actions taken. By adhering to these standards, network operators are well-positioned to achieve and maintain the expected level of quality for end-users, as defined in Service Level Agreements (SLAs). These SLAs often establish demanding performance or reliability requirements, such as achieving "five nines" (99.999%) uptime, which is especially critical for essential systems like power, transportation, and telecom networks.

Security management standards – such as the ISO/IEC 27000 series [6], ETSI TVRA [5], NIST SP 800-53 [9] – distinguish between the business policy topic (i.e. rules for leveraging the information system to detect and track illegitimate activities) and the security implementation topic (i.e. selecting protocols and mechanisms to enforce security). These standards provide guidelines for establishing an Information Security Management System (ISMS), which focuses on risk mitigation through a multilevel

[3] Quoting [2]: the way in which the actual occurrence of a disturbance or fault is analysed with the purpose of providing an appropriate indication of performance and/or detected fault condition to maintenance personnel.

[4] Quoting [4]: "any kind of malicious activity that attempts to collect, disrupt, deny, degrade, or destroy information system resources or the information itself."

iterative approach, such as a plan-do-check-act cycle. Methodological frameworks, such as OCTAVE [10], EBIOS RM [11], and NIST SP 800-61 [12], can help organizations setting up a cybersecurity management organization and aligning it with the ISMS.

The Incident Management Process (IMP) and Incident Response Management (IRM) processes are designed to meet SLAs and security requirements (Figure 2). The terms IMP and IRM can be considered interchangeable or have some level of difference depending on the perspective of the individual [13]. Notably, due to the sensitive nature of cybersecurity incidents, the SecOps incident lifecycle has a natural inclination to be seen as an IRM. However, the underlying concepts apply to both the NetOps and SecOps perspectives. For example, the containment stage in the SecOps context (e.g. disabling a compromised user account to prevent the attacker from accessing endpoints and other resources in the network) is similar to the restoration stage in the NetOps context (e.g. applying a failover activation procedure on a load balancer cluster to prevent the loss of access to an application). In both cases, the concept of an incident (impacting the service or its security) is defined from the perspective of the end user. The decision-making capabilities regarding the actions to be taken for system restoration, attack containment, incident resolution or repair depend on the diagnostic stage. Ideally, a Root Cause Analysis (RCA) leads to a clear diagnosis (i.e. to be able to clearly and unequivocally state which event on which asset is at the origin of the situation), and consequently, an objectively immediate choice of the repair procedure.

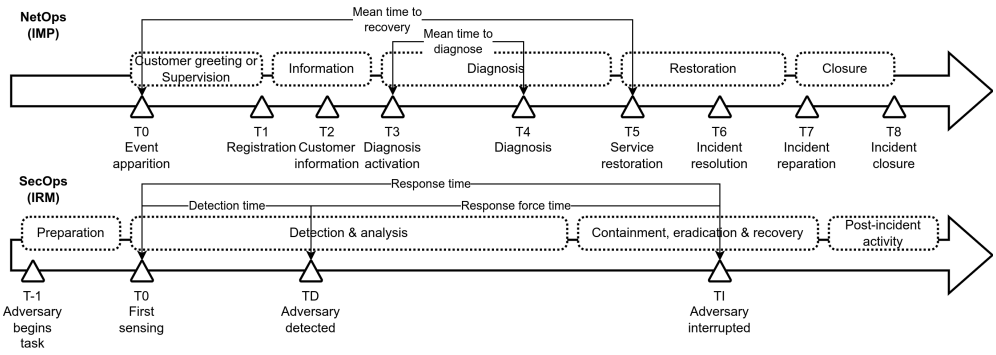

Fig. 2: Incident Management Process (IMP) *vs* Incident Response Management (IRM).

Drawing a parallel between the alert and incident management processes for NetOps (ITIL Incident Management [14]) and SecOps (Management of Information Security and Improvements [15, Annex A 5.24] & Design Basis Threat framework [16]) contexts, showcasing their fundamental stages and temporal milestones. These milestones can serve as metrics for evaluating process performance and also for modeling and analyzing cyber-physical interactions.

3 Challenges inherent in ICT systems operations

In this section, we detail the objectives of this survey by analyzing the challenges inherent in ICT systems operations, and we state our working hypotheses.

Process performance and knowledge capitalization.

To effectively manage network operations, including incident management, it is essential to have a thorough understanding of the ICT systems' state and operational rules. In the absence of such knowledge, it becomes necessary to develop a method to group and prioritize the multitude of indicators and notifications originating from network operations, which are reported through NMS and SIEM systems. These notifications encompass various aspects like system and application logs, performance indicators, technical alarms, and service alarms, which are associated with spontaneous faults, configuration changes, normal usage, and malicious activities.

By implementing a systematic approach to managing these information and notifications, several benefits can be realized. For instance, it would allow to highlight explicit behavioral patterns exhibited by the network, promptly report critical situations, facilitate tractable analysis of incidents, and identify new instances of faults that are identical or similar to previously encountered issues. Moreover, this process could enable the creation of new supervision rules that give priority to relevant notifications until the root cause is pinpointed, thus establishing a connection with appropriate remedial actions.

Scale effect.

As networks grow in size and complexity, organizations tend to adopt siloed structures to maintain high expertise within specialized teams. This dual dynamic leads to Network Operation Center (NOC) and Security Operation Center (SOC) teams becoming overwhelmed with data analysis and alarms prioritization, considering that a thousand of events per day and per security engineer is the practical maximum to deal with [17]. Although automatic alarm prioritization solutions are used to assist, they have limitations: rule-based solutions result in false negatives and are hard to maintain, while machine learning-based solutions lead to false positives and lack explainability.

Another consequence of this dual dynamic is that the knowledge about ICT systems' behavior is fragmented across different teams with varying terminologies and rules, even for similar equipment types. Hence, although the variety of decision support tools and technical solutions to manage networks is a wealth in itself that corresponds to the variety of technical or functional scopes to be managed, it is at the same time a challenge for efficient incident situation understanding: in practice, decision-making on the remediation action to be taken for a given situation must be based on a multiplicity of viewpoints stemming from various specialized tools. This diversity also makes it challenging to consistently and practically account for system behavior, thus undermining the principle of continuous improvement loop recommended in quality management standards like the ITIL Incident Management process [14], ISO/IEC 20000 [6] and NIST SP 800-61 [12].

Knowledge representation.

We observe that graph-based technologies are more and more used to monitor complex systems or help to detect anomalies [18–21]. We posit that using graphs as representation paradigms can be highly beneficial to network and cybersecurity administrators

for improving situation understanding and response through User and Entity Behavior Analytics (UEBA) [22] solutions as part of NMS and SIEM. Beyond the fact that networks are generally represented as graphs for the intuitive value of this representation mode, employing an association model is crucial because similar network events can lead to different incidents depending on the technical context in which they occur (i.e. thinking of network configuration in the broad sense, including network topology and equipment and service parameters). In fact, not using graphs eliminates the possibility of considering an incident event as part of a larger whole that may itself contribute to more complex events/incidents (e.g. common cause failures, cascading failures and alarm spreading phenomenon).

Heterogeneity in concepts and data.

At the same time, using graphs necessitates imposing a certain level of consistency in knowledge representation to ensure effective data utilization. The heterogeneous nature of network data (e.g. technical characteristics of assets, technical logs and alarms, performance measurements as time-series, users and organizations), combined with the fact that graphs are a versatile data structure, may lead to the temptation of directly recording this diversity of data in the graph, potentially overlooking the need for generalization for downstream analysis tools (e.g. interpreting network interface state changes consistently, irrespective of the manufacturer providing the textual notification). Building upon earlier management protocols like SNMP [23], various modeling languages and data models are now available to tackle this heterogeneity challenge. For instance, the TM Forum Open APIs [24] offer interoperable definitions for states and operations in decision support tools, the YANG [25] modeling language describes configuration and state data of network elements, and several prior studies in close relation to knowledge graphs [26] and Semantic Web [27] technologies have shown the value of semantic modeling in network infrastructure monitoring, such as INDL [28], CRATELO [29], UCO [30], ToCo [31], ACCTP [32], and DevOpsInfra [33].

The use of graphs also opens the door to the use of analysis and inference tools directly aligned with graph theory (e.g. risk diffusion using shortest path calculations [34], event clustering using a centrality measure [35]), machine learning (e.g. fraud analytics using graph isomorphism [36]), or automated reasoning (e.g. identifying computer forensic scenarios using pattern matching [37]). However, introducing new AI techniques or adapting existing ones for anomaly detection and root cause analysis within graph structures must prioritize compatibility with this data format. Challenges may arise, such as handling dynamic graphs [35, 38] or data transformations with information loss at the models' input, thus leading to costly inference pipelines for result recontextualization. Similarly, relying on a single AI technique may prove inadequate for addressing the wide range of failure or attack scenarios that can arise. Typically, detection tools and models specialize in specific data types and detection cases, like using file excerpts to detect viruses by their signatures [39] or IP packet headers to identify trend disruptions in application usages [40]. However, incident characterization often involves multiple artifact types or compromise indicators due to the complex nature of large-scale system failures and the ingenuity of malicious actors in devising attack scenarios. Therefore, unless an inference model capable of

handling all data types and detection cases is available, an architecture like synergical reasoning [41] (cooperative decision making) becomes essential. Such an architecture could follow a model stacking scheme [42], or incorporate diverse specialized inference models that collaborate, leveraging each other's outcomes through mechanisms like voting, on-demand inference and re-use of previous inference results.

Observability and decision-making.

Finally, since ICT systems are inherently interconnected, any action of modification (whether it is intended to evolve or repair the system) must be taken with full knowledge of the consequences. Therefore, having precise and complete knowledge of the system is a desirable ideal, but it is not the general case for various reasons. From a completeness perspective, typical examples include the absence of measurement means, encrypted or protected data, or dropped notification data unit by the network. From a precision perspective, the typical case is the accumulation of layers of data interpretations, each introducing errors that hinder unambiguous decision-making. Taking inspiration from [43], this phenomenon can typically be represented with Eq. 4:

$$D_T = T_H(T_S(T_P(D))) \tag{4}$$

where T is an interpretation function, D is the data representing a system state, measured by a probe P, encoded into the information system S, and then understood by a human operator H for potential decision-making.

The first drawback arising from this observability context is that the diagnostic phase corresponds to a special situation of making an inference from vague or fuzzy premises [44], thus necessitating the inclusion of a belief or confidence indicator with the results of the RCA (Figure 3). The second drawback is the risk of error in choosing

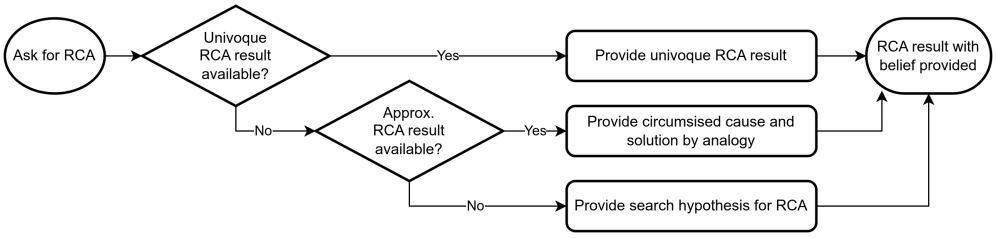

Fig. 3: Graduated Root Cause Analysis (RCA) computation strategy.
During incident diagnosis, the accuracy of root cause search results determines remediation action selection. In the best case, actions are deduced from the cause; otherwise, abductive reasoning [45, 46] is used to select probable causes/solutions or present research hypotheses.

the remediation action. This is theorized in [41] through equation $L \equiv C \wedge P_i \rightarrow G\langle p \rangle$, where L is a logical lemma, C a concept, P_i the i^{th} procedure stored in a knowledge base, G the goal to be reached (i.e. returning the system to normal operations) and $\langle p \rangle$ a probabilistic measure of confidence over L.[5] The typical approach in this kind of

[5] The equation can be read as "If the context C appears to hold currently, then if I enact the procedure P, I can expect to achieve the goal G with certainty p" [41].

situation is to proceed through trial and error until the system is brought back to a viable state. This involves making decisions in uncertainty about a system whose state evolves over time, influenced by its own dynamics and the consequences of attempted actions for incident recovery (Figure 4).

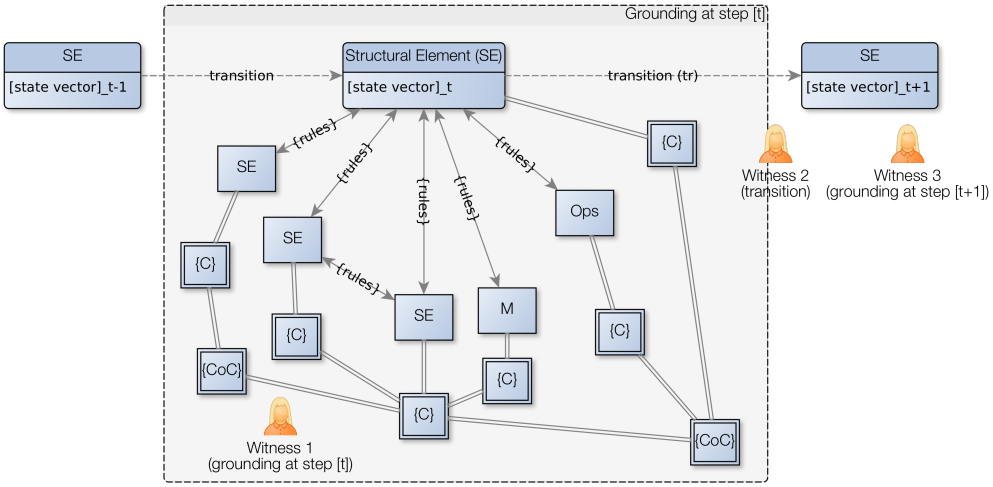

Fig. 4: Sequential & uncertain decision problem on a hybrid "concrete-conceptual" model.
Witness 1 interprets the network's state (a set of state vectors related to structural elements SE at time t) abstractly using concepts (C) and combinations of concepts (CoC). Each time step has potential future states based on the technical and conceptual neighborhood of SE (M = malicious actor, Ops = planned operations, etc.), and a state transition model (set of $rules$). Witness 2 can predict future state vectors by analyzing transitions or actions on the system, while Witness 3 observes the system with its new set of states and concepts (same as Witness 1) resulting from the application of the state transition model. Lack of knowledge about state vectors or rules leads to considering the inference process (e.g. alerting on undesirable user/system trajectory, predicting next user/system action for corrective maintenance) as a sequential decision-making problem under uncertainty, where states and transitions represent the system's dynamics.

Working Hypotheses.

Considering the aforementioned challenges, we aim to provide a constructivist framework towards mastering the complexity of ICT systems behavior in the context of network operations and incident management. Therefore, we propose the following main assumptions or working hypotheses for guiding this survey.

Hypothesis 1. The fundamental model for representing ICT systems and their dynamics is a dynamic graph (network topology + notifications = dynamic graph).

Hypothesis 2. Alarm spreading and cascading failures are bounded in terms of both time and location.

Hypothesis 3. The functioning logic of ICT systems can be (partially) inferred or learned by observing both network topology and notifications.

Hypothesis 4. The state trajectory of ICT systems reflects the course of action, including malicious usage.

Hypothesis 5. The state trajectory of ICT systems can be mapped to a logic-based abstracted representation of the situation.

Ultimately, by leveraging diverse data sources and drawing insights from domain experts, the research community, and shared/private data models, the framework based on these hypotheses could facilitate understanding the behavior of a complex technical system. This understanding could then be used to reason about the system's trajectory in a transparent manner through a fundamental data flow abstraction (Eq. 5), enabling applications such as anomaly detection and optimal design calculations.

$$\mathcal{E} \xrightarrow{e.t.l.} \mathcal{K}^{\circlearrowleft^r} \xrightarrow{infer.} \mathcal{P} \tag{5}$$

where $\mathcal{E}$ is the Environment (i.e. primary and secondary data describing the network)[6], $\mathcal{K}$ is the Knowledge (i.e. information about the network behavior and business rules guiding network administration tasks), $\mathcal{P}$ are the Propositions (i.e. explained inferences about the network states and behavior), $e.t.l.$ is an Extract-Transform-Load process or alike, r is a reasoning process (i.e. an "internal" inference process aimed at producing facts and knowledge from basic facts present in $\mathcal{K}$) and $infer.$ an inference process.

4 Research Methodology

In this section, we explain the methodology used for creating this survey. Considering anomaly detection for ICT systems as an end-to-end research topic, we followed a two-step research methodology to collect and analyze relevant scholar and technical articles.

Firstly, in relation to the challenges discussed in Section 3, we broke down the field into the four following research axes to establish a framework for exploration and analysis: **Knowledge Representation (KR)** as the capability to store and process heterogeneous data; **CompleXity (CX)** as the capability to (efficiently) handle and process data streams as well as older/static stored data; **Anomaly Detection (AD)** as the capability to contextualize events and use them as a basis for complex anomaly detection; **eXPlainability (XP)** as the capability to provide feedback to the decision support system users about the reasoning process that led to a given alert.

Next, we conducted a keyword/topic-based systematic bibliographic research across various sources including peer-reviewed scholar venues, standardization bodies and well-known organizations such as Gartner or CISCO, filtering on the targeted application domains (e.g. networks and data centers management, software engineering, cybersecurity, cyber-physical systems). We also considered the co-citation network, adoption degree, and the availability of implementations. In order to address the subject comprehensively, our initial set of keywords included "*anomaly detection*", "*failure detection*", "*availability*", "*dependability*", "*self-healing*", and "*process modeling and comparison*". Then, regarding the knowledge representation axis, we further looked for data structures and knowledge representations commonly used in the ITSM and cybersecurity domains with more specific keywords: "*network and system*

[6] We use a general definition of *primary data* (data directly collected on the source, may be in raw format or that have been normalized for future processing steps) and *secondary data* (data that has already been collected, processed, or even aggregated).

topology/functional description", "*event/fault/signaling notification*", "*asset management*", "*communication protocol definition*". This included scrutinizing telco and IT standards, user manuals of products, and code analysis from the research community projects. As graphs have a natural affinity with ICT system diagrams, we primarily targeted knowledge representations with native support for graph structures, notably knowledge graphs [26] based on Semantic Web technologies [27] for their knowledge interoperability and inference capabilities. We also expanded our exploration to more traditional data structures such as relational tables in database management systems, decision trees, and Bayesian networks [47], as translations between these structures and graphs can be made back and forth with help of additional tools.

Using this method, we have selected ~ 270 references that have been further analyzed according to the specific criteria for the three sections that follow: in Section 5 (capabilities of DSSs), we examined a corpus made of approximately 65 industrial tools, file formats, and data exchange standards; in Section 6 (semantic models), we examined 97 references that were published between 2004 and 2025; in Section 7 (algorithmic methods for anomaly detection), we examined 106 references that were published between 1999 and 2024. Note that some references may address more than one research focus. For example, FOLIO [48] describes the development of knowledge-based anomaly detection techniques that is typically associated with the development of a semantic model that allows for representation and reasoning in the discourse domain.

5 Detecting Anomalies and Malicious Activity: a Cartography of Industrial Tools

Designing a data processing architecture for incident management of ICT systems involves various research and technical domains, such as data transformation and wrangling, storage and processing architectures, decision making, and business process management. In this section, we review related work focusing on the current architectures of Network Monitoring System (NMS) and Security Information and Event Management (SIEM) systems. First, we examine high-level requirements from the eye of NetOps and SecOps in Section 5.1. Then, we analyze the well-established capabilities of these tools in Section 5.2. Finally, we discuss the remaining limitations in Section 5.3.

5.1 What Do Experts Need To Ask Monitoring Tools and Decision Support Systems?

As mentioned in Section 2.2, recommendations apply to both the NetOps and SecOps domains in terms of organization and tooling. These recommendations aim to achieve the dual objective of maximizing service quality/security and minimizing anomaly detection/correction times. In both domains, the incident management process is described as a sequence of iterative steps including the diagnosis of the situation and leading to the remediation and correction of an undesirable situation. As such, it is akin to an "action-observation-reward-goal" process model [49] with the following scenario:

1) a failure (issue) on an asset induces events and alarms on the asset's neighborhood; *2)* responding to a trouble ticket (an alert), a network or security administrator analyzes events and alarms to distinguish primary events (causes) from secondary events (effects); *3)* contextualizing events and alarms with respect to "in policy" or "out of policy" activity models enables the administrator to select a remediation action; *4)* based on the remediation action results, the administrator closes the trouble ticket (the issue) or loops back for further analysis and corrective actions.

When engaged in this four-step scenario, experts wish to get an accurate insight about some specific event occurring over the network and be able to answer fundamental questions such as: "what are the objects involved?", "what are the observations?", "what can we infer from a set of observations and why can we infer this?", "what are the causes / consequences?" and "what is the remedy?".

It is noteworthy that these questions follow an incremental situation understanding task scheme, i.e. from elementary queries on a knowledge base (e.g. using SPARQL queries [50], the DROOLs [51] engine) to queries combined with specialized inference modules (e.g. rule-based, entailment [52], classification, what if model [53], digital twin simulation [54, 55]). It is also noteworthy that these questions lead to secondary requirements in terms of reasoning capabilities upon situations: *dependency calculus* (e.g. what services are at risk whenever this host fails?); *causality inference* (e.g. is this log related to some other log?), *situation awareness* (e.g. is this set of log anecdotal evidence of an attack course of action?). These questions and remarks themselves are guides for the expectations of NetOps and SecOps teams regarding monitoring tools and DSSs.

5.2 What Are the Well Established Capabilities of Those Tools?

Umbrella systems for centralized situation understanding.

To support network and security administrators in their task, numerous tools and procedures are available for diagnosing the state of ICT systems (remote access to devices [56], on-site measurements and indicators from probing systems [57], decision support tools such as NMSs [58, 59] or SIEMs [60, 61]), monitoring the life cycle of incidents, and capitalizing on knowledge of the causes and solutions to incidents (help desk ticketing systems [62], knowledge bases [63]). This variety of tools and solutions is a wealth in itself that corresponds to the variety of technical or functional scopes to be managed (e.g. network traffic analysis [64, 65] *vs* malware signature in files [39], IPoDWDM international backbone network *vs* distribution and access data center network). However, this is at the same time a challenge for a unified approach to the diagnostic stage: in practice, decision-making on the remediation action to be taken for a given situation must be based on a multiplicity of viewpoints stemming from various specialized tools. NMS and SIEM platforms play a crucial role in addressing this challenge, as their typical role is to centralize and present all technical information and alerts from networks.

Log recording/management and notification analysis.

NMSs and SIEMs are two different product lines due to the nature of the data processed and the expectations regarding the incident management processes in which they are involved. For telecommunication networks, alarms (i.e. a durable or non fugitive fault that happens on an atomic function, as discussed in Section 2.2) are first class citizens that should be reported to a MCS for performance analysis and service impairment detection. For cybersecurity, technical logs need to be combined with vulnerabilities and threat intelligence in a Log Collection → Log Normalization → Notifications and Alerts → Security Incident Detection component chain for threat response management [17]. Both product lines show two main functional blocks: log recording/management and notification analysis, with the second block dependent on the first. Figure 5 illustrates this for SIEMs, highlighting the characteristic sub-functions of each block. It also shows that the implementation of the recording/management block primarily involves technical solutions, while the analysis block primarily involves algorithmic solutions.

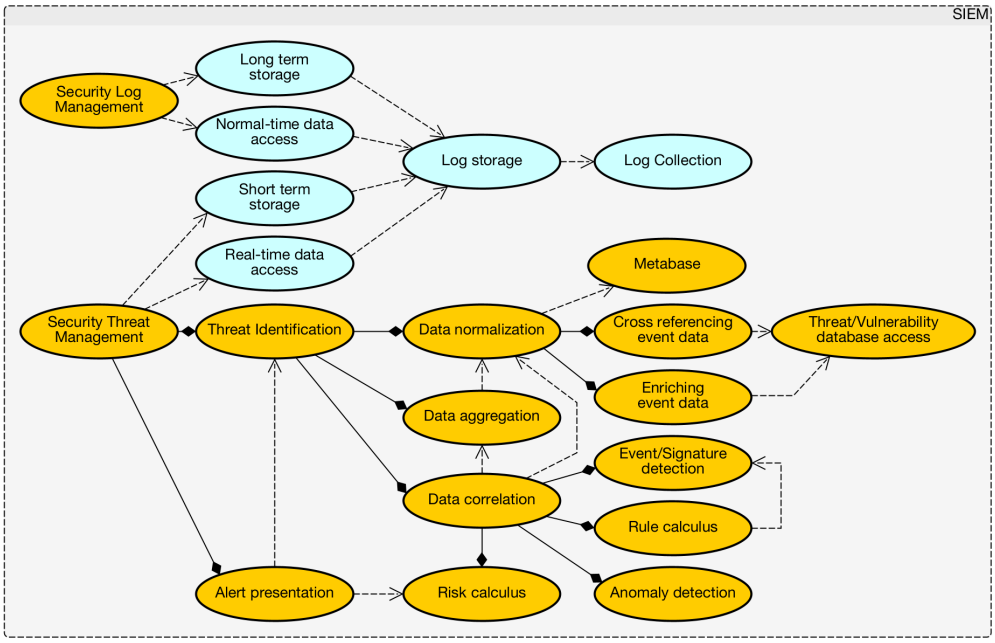

Fig. 5: SIEM technical and functional capabilities.
Use case analysis for SIEMs, based on [17], using the UML representation standard. Dotted lines indicate a functional dependency, and diamond lines represent a composition relationship. The light blue use cases (top ellipses) are technical-focused use cases, while the other use cases are primarily algorithmic-based.

Handling heterogeneous data and short/long-term analysis.

As ICT event data falls into the category of big data (e.g. high volume, variety, and velocity characteristics) [66], design choices for the implementation of the recording/management and analysis features are influenced by the need for efficient data

ingestion, processing performance, and data retention. These design choices aim to enable both near-real-time anomaly detection (e.g. event correlation indicating the propagation of malware) and long-term behavior analysis of the systems (e.g. calculation of the characteristic propagation method of a given malware).

In both NMS and SIEM contexts (e.g. ZenOSS [67], NetWitness [68]), data processing architectures generally follow the producer/consumer design pattern (a.k.a. observer pattern [69]) and inspirations from distributed computing (i.e. hubs + aggregator).[7] This type of architecture indeed offers a level of modularity that allows for managing the integration of data sources with varied persistence and dynamics characteristics through specialized modules that operate independently. This is particularly the case when monitored systems are heterogeneous, such as a High-Performance Computing (HPC) platform with a data transfer and processing service offer *vs* an Internet service provider with communication service and Platform as a Service (PaaS) offers.

The architectures used also allow for combining multiple storage solutions. The general trend is to maximize Input/Output (I/O) performance and minimize storage footprint, subject to both data semantics and persistence/dynamics characteristics: *1) daemons and web applications* (e.g. dedicated filesystem for raw logs, binaries and libraries); *2) events and node information* (e.g. PostgreSQL[8] for structured notifications and characteristics); *3) performance data* (e.g. RRD[9] for throughput or CPU usage time series). When not combining storage solutions, alternative approaches are to apply rule-based data normalization before storage (e.g. ManageEngine's EventLog Analyzer[10]), store raw logs/events then normalize data before applying rule-based analysis modules (e.g. SolarWinds's Security Event Manager[11]), or apply in-memory real-time analysis of data streams and process the resulting information (e.g. OSSEC Foundation's OSSEC HIDS[12], IBM's QRadar[13]). It is noteworthy that durable storage formats mainly correspond to relational database management systems or time series databases, rather than graph formats, except for new entrants in the market (e.g. Luatix's OpenCTI[14], EXFO's Nova Context[15]). Overall, the proposed approaches are a compromise between two related trends: how data is conceptually stored and managed, and how hardware/software handle data.

Information correlation and shared services.

Log centralization in NMS and SIEM systems allows network administrators to focus their monitoring and analysis activities on a single tool. An additional strategy implemented by monitoring tools and DSSs to minimize (cognitive) resources required for analyzing alarms and logs when presented to the operator is based on the concept of

[7] The TM Forum's Open Digital Architecture (ODA) aims to improve user experience and Information System (IS) interoperability in the ICT industry beyond general best-practice approaches for Decision Support Systems design.
[8] https://www.postgresql.org/
[9] http://www.rrdtool.org/
[10] https://www.manageengine.com/
[11] https://www.solarwinds.com/
[12] https://www.ossec.net/
[13] https://www.ibm.com/qradar
[14] https://www.opencti.io/
[15] https://www.exfo.com/

semantic distance (i.e. the distance between the goal aimed by the user and the actions/objects of the user interface [70]). Various complementary approaches are observed in this regard, among which *notification contextualization through rendering* and *notification rewriting and enriching* are playing a significant role and typically leverage the DSS operators' skills through implicit characterization of the notification. Regarding rendering, classic examples include flashing a shape on a network map and displaying the alarm in text form alongside other information related to the equipment or service affected by the alarm. For rewriting and enriching, examples include mapping an alarm to a basic supervision category[16] and annotating it with `probableCause` [3] attribute value or confidence score based on inference services (Figure 6), thereby relating the notification to the fault interpretation domain and the context for assessment.

Another strategy involves *grouping and hierarchically prioritizing notifications* to facilitate RCA and decision-making for remediation actions. RCA for alarm spreading phenomenon in ICT systems is typically approached as an inductive process that distinguishes between primary failures (i.e. a failure that directly indicates the fault location and initiate a repair action, e.g. a broken cable or a misconnection) and secondary failures (i.e. a consequential failure, e.g. an upper level service that is gone down) [2, Section 7.1.1.1]. Therefore, the cause of a data transmission impairment is sought in the first alarming network element of a datapath, and redundant Alarm Indication Signal (AIS) can be silenced using an alarm suppression function or linked to a parent notification using a `correlatedNotifications` [3] attribute.

Similarly, for non-hierarchical or more complex systems, UEBA is typically approached through a doubt removal process using dependency graphs or decision trees. Applied examples of this model-based approach are present in NMSs/SIEMs where it is assumed that the network topology and transaction records (i.e. a sequence of one or more operations – reads or writes – which reflects a single real-world transition [75]) are a model for causality relationships about network elements, services and applications. As this kind of feature is a key differentiator for commercial tools, few detailed papers are freely available about this but online blog posts for demonstration purposes. For examples, we refer to the Zenoss "Layer 2 ZenPack" [76], the Riverbed "APM" [77] or the Cisco AppDynamics "Cognition Engine" [78]. In the absence of algorithmic solutions, doubt removal is typically achieved in DSSs by facilitating an exploratory approach to data through the use of hyperlinks for navigating between information elements, display filters, a query system on the DSS database, and even interactive annotation (e.g. IBM "i2 Enterprise Insight Analysis" [79]) and event sharing among the supervision staff.

5.3 What Are the Remaining Limitations?

Although NMSs/SIEMs are a data focal point for network element characteristics and operational states, it is noteworthy that automated contextualization of notifications does not (or minimally) take into account network topology information (e.g. network

[16] As per [2, Section 7.1.1] for telecommunication networks: transmission (management of the transmission resources in the network), quality of service (degradation in the performance), processing (software or software processing fault), equipment (fault localization and repair of the equipment itself) and environment (ambient conditions within an enclosure in which the equipment reside). In the cybersecurity domain: Confidentiality, Integrity, Availability (a.k.a. the CIA triad).

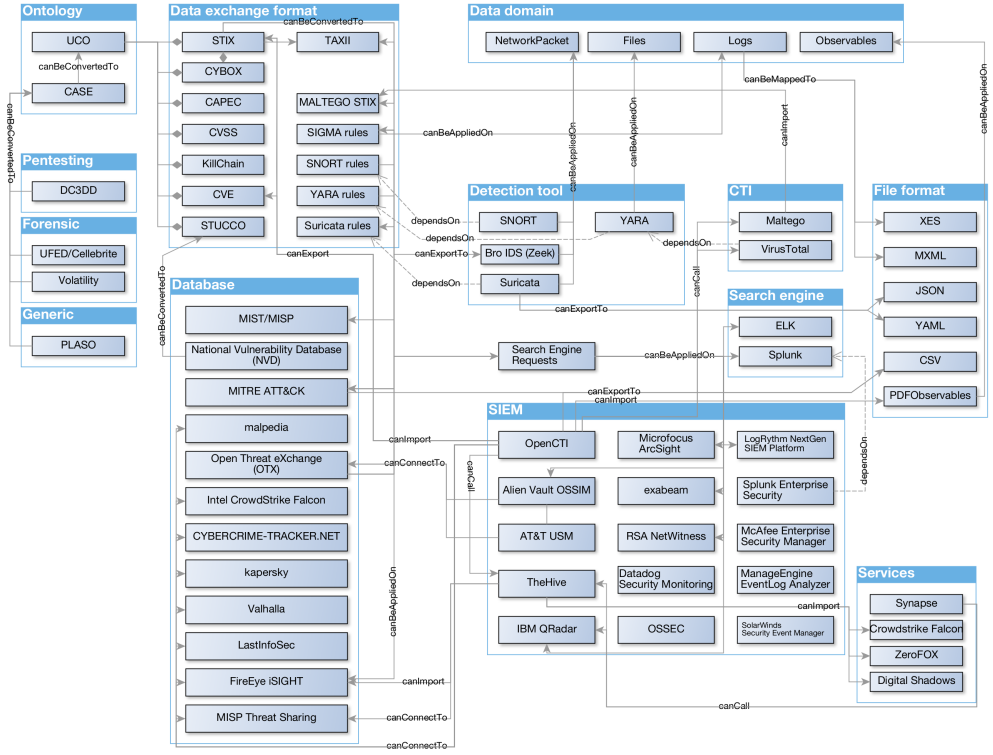

Fig. 6: Cybersecurity tools relationships.
This entity-relationship diagram shows how specialized cybersecurity applications like SIEM (e.g. OpenCTI) or Cybersecurity Threat Intelligence (CTI) tools (e.g. VirusTotal) can compose, with help from shared services (e.g. NVD [71], MISP [72], Maltego [73]) or exchange formats (e.g. STIX [74]), in order to provide contextualized information over detection tools (e.g. SNORT [64], YARA [39]). Regarding relationships around SIEMs, emphasis is placed on the OpenCTI tool due to the easy accessibility of its detailed features, as it is an open source tool. Relationships around the UCO [30] component showcase the data sources and exchange formats used for knowledge discovery and After Action Report (AAR) annotation tasks as mentioned in [30], providing insights into the functional domains covered by the UCO project. The method used to construct this survey consisted of distinguishing, within the exchange formats, the aspects from those that rely on specific analysis tools, and then tracking the interrelationships between these tools.

links, data flows, routing tables, failover mechanisms, location), network operation information (i.e. current and past trouble tickets or scheduled operations), agreed-upon work methods (e.g. equipment upgrade or IP address blacklisting procedures), or information regarding services provided to users (e.g. contractual data, business functions affected by a network service). Instead, it is rather the principles of User Interface/User eXperience (UI/UX) ergonomics and easy access to complementary (ad hoc) tools (e.g. search engine of Operations Support Systems (OSS), network diagram inference from traffic dumps, queries to third-party CTI tools) that are deployed, trusting that the DSS user will have and know how to mobilize the necessary skills to perform diagnostics.

Part of the reasons for this state of affairs comes from the fact that networks are complex and open systems (i.e. no *a priori* knowledge of the behavior of users and connected neighboring systems) are constantly evolving (e.g. new customers, new devices, new services). This, in turn, leads to an administrative effort in managing DSSs, defining alert rules, and ensuring data coherence, with costs exceeding regular operational

expenses. For example, regarding the RCA and UEBA techniques evoked in Section 5.1, it is indeed necessary to consider that learning and using causal models rely on the idea of having complete knowledge of the ICT systems. This requires a complex technological ecosystem composed of network discovery protocols (e.g. LLDP [80]), cross-vendors definitions of managed objects (e.g. non vendor-specific branch in a SNMP Management Information Base or in a YANG model [81]), active network monitoring systems (e.g. flow monitoring with an IPFIX [82] compliant system) running over an in-production network, and even information sharing between network production and operation stakeholders. Regarding information sharing, functional alignment issues can be observed between the data models and vocabularies implemented and used by network production and operation stakeholders, thus entailing poor interoperability (e.g. challenges in interpreting potentially similar facts and concepts) and further complicating the implementation of RCA and UEBA techniques (i.e. causal relationships filtering for unseen yet failure modes or incident situation is generally unavailable unless explicitly implemented by NetOps and SecOps experts responsible for a specific network). Table 1 illustrates this with data exchange formats in the field of cybersecurity, along with the existence of numerous cybersecurity taxonomies from various sources: European Commission [83, 84], NIST Computer Security Research Center[17], IEEE[18], IFIP[19], ECSO[20], cyberwatching.eu[21], and so on.

Although interoperability of data representations and exchange formats is a cornerstone of current limitations, the heart of the problem lies in the ability of DSSs to effectively support the diagnostic steps performed by experts by providing full and reliable contextualization of events occurring on the ICT systems. According to Cybernetics [86], processing a notification alone cannot increase its informational content, but enhancing the precision of the knowledge about the situation in which it occurs can be valuable, even in ambiguous situations. Similarly, time psychomechanics [87] suggests that a complete understanding of an object involves considering not only its current state but also the various states it has gone through. Based on these principles, a natural approach to enhance the capabilities of NMSs/SIEMs is for analysis algorithms to rely on a comprehensive and integrated view of ICT systems and their ecosystem, rather than aggregating inference results from several algorithms, each using a subset of data. The means to implement this proposal are analyzed in the following sections, starting with the knowledge representation and reasoning perspectives in Section 6, and then the anomaly detection perspective in Section 7. Ultimately, as insights from NetOps can benefit SecOps and vice versa, NMSs and SIEMs could then be considered as part of the same DSS solution.

[17] https://csrc.nist.gov/
[18] https://standards.ieee.org/practices/foundational/cybersecurity-standards-projects/
[19] https://www.ifipsec.org/
[20] https://ecs-org.eu
[21] https://www.cyberwatching.eu/

Table 1: Cybersecurity vocabularies and featured application domains.

	Asset Definition	Configuration Guidance	Vulnerability Alert	Threat Alert	Indicator Sharing	Incident Report	Configuration Guidance Analysis	Vulnerability Analysis	Threat Analysis	Intrusion Detection	Centralized Reporting	System Assurance	System Assessment
CAPEC				✓					✓	✓	✓		✓
CCE		✓					✓	✓	✓	✓	✓		
CCSS		✓					✓				✓		
CEE					✓	✓				✓	✓		
CPE	✓				✓	✓	✓	✓	✓	✓	✓	✓	
CVE			✓	✓	✓	✓		✓	✓	✓	✓		
CVRF			✓										
CVSS			✓	✓		✓		✓	✓	✓	✓		
CWE			✓	✓		✓		✓	✓	✓	✓		✓
CWSS						✓		✓			✓		
CYBEX						✓							
CYBOX				✓	✓	✓			✓	✓	✓		
IODEF					✓	✓							
MAEC				✓	✓	✓			✓	✓	✓		✓
OCIL		✓					✓	✓	✓	✓		✓	
OVAL	✓	✓	✓			✓	✓	✓	✓	✓	✓	✓	
SBVR STIX				✓	✓	✓			✓	✓			✓
SWID	✓						✓	✓	✓	✓	✓		
XCCDF		✓					✓	✓	✓	✓	✓	✓	

Comparison of the application domains for well-established cybersecurities vocabularies, based on [85].

6 Knowledge Representation and Reasoning: a Cartography of Semantic Models

In order to represent data in graphs for NMSs and SIEMs, the remaining limitations discussed in Section 5.3 highlight the need to combine various facets of knowledge. Ontologies – as explicit representations of a discourse domain through concepts and relationships – and their instantiation as knowledge graphs [26], enable data analysis and inference techniques to handle heterogeneous data and reason about the context of represented objects. In this section, we provide a summary of the ontologies we have collected during our systematic literature review, starting with the definition of evaluation criteria in Section 6.1, and reporting on our analysis in Section 6.2.

6.1 Evaluation Criteria

The data collected and evaluation criteria used to analyze the ontologies are as follows:

- **Name.** The name of the ontology or, if not available, the title of the document describing it (e.g. research paper, specification, website).
- **Primary application domain.** The domain or field of application that originated the ontology, based on the indications provided by the authors, or if not available, deduced by us.
- **Bibliographic document category.** We categorize reference documents discussing the data model based on a detailed reading to estimate the effort required for understanding, using, or implementing the model. The six categories are: "*position*" (analyzing domain issues and expressing intentions for future work), "*overview*" (providing a high-level description and use cases), "*specification*" (formally describing the design methodology and model), "*dataset*" (primarily presenting a dataset), "*documentation*" (providing context and usage information), and "*no access to paper*" (due to broken links or access restrictions).
- **Main concepts and relations.** We sample concepts and relationships from the data model, either from its implementation or the authors' description. We focus on top-level concepts and their relationships, and sometimes second-level concepts if there are few top-level ones. In complex models with many top/second-level concepts and shallow hierarchy, we consider the centrality of concepts. This centrality indicates their importance in the domain's conceptualization and can be inferred from graph diagrams in ontology reference documents (research papers, documentation).
- **Availability and location.** We assess availability of data model implementation: "*yes*" (publicly accessible), "*broken link*" (unavailable despite reference), "*empty content*" (reachable but empty), "*no reference provided*" (location not indicated or explicitly stated as unpublished), or "*not relevant*" (not related to Semantic Web or data model). If implemented, we record URL and serialization used.
- **Bibliographic citation count.** We evaluate the usage or consideration of the proposed data model based on the number of citations of the research paper or reference document describing it. The citation count is obtained from Google Scholar[22], and the corresponding URL is recorded.[23]
- **Ontology metrics.** We gather the characteristics of the data models, including the number of classes, object properties, data properties, individuals, and Description Logic [52] expressivity. This information is collected using the Protégé 5.1 tool [88] or, if we do not have access to the implementation of the model, based on the details provided by the authors (in research papers or online documentation describing the data models).
- **Best practice compliance.** We assess the quality of implemented data models using best practices described in [89] and the related OOPS! tool[24]. To do this,

[22] https://scholar.google.com/

[23] To fully understand ontology adoption, detailed usage information such as the number of instantiated classes and its evolution over time would be valuable. However, this information is generally not available.

[24] https://oops.linkeddata.es/

we obtain a copy of the data model (from references in research papers, online documentation, or by contacting authors), normalize it into RDF/XML format using Protégé 5.1, send it to the OOPS! Web service for analysis, and aggregate the results. For modular models, we merge the ontologies into a single file using Protégé before analysis with OOPS!.

- **Conceptual facets coverage.** We analyze the capability of data models to encompass the four facets of the discourse domain – as established in [90] by considering dynamic ICT systems with constrained and multi-level functional behavior (Figure 7) – that are essential for representing networks and their dynamics: *structural* (network assets such as servers and links), *functional* (network services and flows), *dynamic* (events and states changes) and *procedural* (processes and actions). We evaluate this by examining main concepts and relations of the data models using a subset of competency questions (Table 2) derived from NetOps and SecOps expert panel interviews conducted in [90].[25]

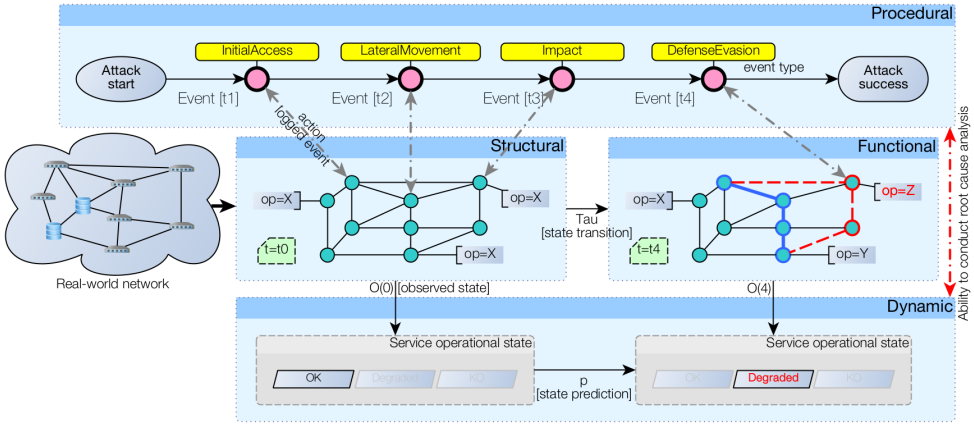

Fig. 7: ICT system state transition model and four-faceted knowledge domain.
The representation of a network can be divided into four facets: *structural*, *functional* (the blue path indicates an operational data flow, the red path a faulty flow), *dynamic*, and *procedural* (logged events are related to cybersecurity attack tactics from the MITRE ATT&CK matrix [92]). Tau stands for state transition, $O(t)$ for observed state at time t, and p for state prediction. Figure and caption from [90].

6.2 Semantic Models

In this section, we present the identified data models from the literature review, their categorization, and evaluation based on the previous criteria. Out of 99 references analyzed, 52 had an implementation using Semantic Web technologies [27] (i.e. using a RDF-related serialization [93] or OWL functional syntax [94]), while the remaining 47 did not have an implementation. In the following, we focus on the models with available implementations. The complete analysis and resources are open source and can be found at https://w3id.org/noria/docs/sota.

[25] We acknowledge that automatic topic modeling could help qualifying a data model for facets. However, based on our experience (e.g. using the ZeSTE tool [91]), we find that it is possible but not reliable due to its dependence on concept naming and description, which is influenced by ontology author bias.

Table 2: Competency questions for analyzing the conceptual facets coverage.

St.	Fu.	Dy.	Pr.	Competency Questions
✓	✓			What assets are shared by a given asset chain?
✓		✓		Which entity (resource/application/site) is concerned by a given incident?
✓		✓		On which resource did this sequence of events take place and in which order?
		✓		What corrective actions have been carried out so far for a given incident (who, what, where)?
			✓	What interventions were carried out on this resource that could have caused the incident?
			✓	What operation plan (automations, operating procedures, etc.) could help us solve the incident?
			✓	Given all the corrective actions carried out so far for the incident, what possible actions could we still take?

The four knowledge facets to represent (St.: structural, Fu.: functional, Dy.: dynamic, Pr.: procedural) map to a subset of competency questions (i.e. user queries expressed in natural language) derived from NetOps and SecOps expert panel interviews, as described in [90]. This panel consisted of 16 experts from Orange[26], an international network infrastructure and service provider, who collectively represent 150 operations team members.

Table 3 provides a list of data models. We observe that the models cluster into six primary application domains (theme), with varying proportions of available models and model characteristics. Table 4 summarizes to what extent the set of models for each application domain theoretically aligns with the four-faceted targeted discourse domain, as defined in Section 6.1 and illustrated in Figure 7. From the proportion of models with identified facets in Table 4, we observe that the models developed in the domains of Process modeling and Smart Environment and Smart Industry (SE-SI) are the most represented. This suggests the significance of modeling efforts in representing networks and their dynamics in these domains. However, the models in these domains generally exhibit disparities in terms of the targeted concepts, as revealed by the fact that process mining models focus on the *dynamic* and *procedural* facets without significant overlap with the *structural* and *functional* facets, which are primarily addressed by models in the SE-SI domain. Similarly, the proportion of models simultaneously covering multiple facets (i.e. $Fx\%$ columns in Table 4) indicates that models in the CyberSec domain seek to cover the maximum of facets simultaneously. This suggests a greater expressiveness and complexity of this group of models (e.g. in terms of the number of concepts and relationships between them).

In this line of thought, regarding the low coupling between facets and potential difficulties in precisely allowing for reasoning on the interplay between network architecture and its operation, a detailed analysis of the data models reveals various opportunities for enrichment, improvement, or linking of the models. For example, the combination of the SEAS and PEP [117, 136] models enables the representation of communication links between technical assets and facilitates the analysis of assets' state changes by tracking commands and results. However, SEAS mainly targets the Internet of Things (IoT) domain and end-user devices, and the semantics of PEP relates to computer process. The DevOpsInfra [33] ontology enables the provisioning of data processing services and tracking computer resources hosting capabilities.

Table 3: Semantic models comparison table.

Theme	Short title	Ref.	St.	Fu.	Dy.	Pr.	Expressivity	CC	OPC	DPC	IC
CyberSec	ACCTP	[32]				✓	ALEROI+(D)	117	52	20	160
CyberSec	CASE	[95]			✓	✓	AL(D)	17	3	7	11
CyberSec	D3FEND	[92]	✓	✓	✓	✓	SROIQ(D)	1568	198	41	1740
CyberSec	ICAS ontology	[96]	✓	✓	✓	✓	SROIN(D)	559	385	144	256
CyberSec	ICS-SEC	[97]				✓	ALHI(D)	56	66	139	0
CyberSec	MALOnt	[98]				✓	ALH(D)	76	11	13	265
CyberSec	ORD2I	[99]	✓	✓	✓	✓	SHOI(D)	67	49	82	14
CyberSec	OWASP OdTM	[100]				✓	ALCROI(D)	100	42	4	30
CyberSec	SLOGERT log extraction ontology	[101]	✓	✓			AL(D)	5	6	4	98
CyberSec	SLOGERT log event ontology	[101]			✓		ALH(D)	12	15	45	0
CyberSec	ForensicOntology	[102]	✓	✓	✓	✓	SHIQ(D)	483	148	51	1800
CyberSec	UCO	[30]	✓	✓	✓	✓	SRIN(D)	422	177	574	455
Generic	A Core Pattern for Events	[103]			✓		ALERI	5	7	0	0
Generic	Basic geo Ontology	[104]			✓		SROIQ(D)	61	22	1	4
Generic	EmOCA	[105]				✓	AL(D)	18	4	3	1
Generic	Event Ontology	[106]			✓	✓	ALCHI(D)	8	17	5	2
Generic	EventKG	[107]			✓	✓	AL(D)	10	3	2	2
Generic	FARO	[108]			✓	✓	ALRI+	5	32	0	4
Generic	FOAF	[109]					ALCHIF(D)	22	36	15	0
Generic	GeoSPARQL	[110]					ALCH(D)	9	37	18	0
Generic	Geonames	[111]					ALEROIN+(D)	15	33	17	701
Generic	Mining Minds Context Ontology	[112]					ALCF(D)	52	4	2	0
Generic	Ontology of units of Measure (OM)	[113]					ALCON(D)	815	23	11	2242
Generic	OWL-S	[114]		✓		✓	ALCHOIN(D)	86	88	41	13
Generic	OWL-Time	[115]			✓		SROIN(D)	21	33	25	15
Generic	PROV-O	[116]			✓		ALCRIN(D)	31	44	6	1
Generic	Procedure Execution Platform (PEP)	[117]			✓	✓	ALCRIF(D)	6	10	2	1
Generic	QUDT	[118]			✓		SHOIQ(D)	138	132	102	78
Generic	Semantic Sensor Network (SSN)	[119]		✓	✓	✓	ALRIN	39	34	0	1
Generic	Vocabulary Status Vocabulary (VSV)	[120]					n.a.				
Health Science	HeLiS	[121]				✓	ALCIQ(D)	281	23	31	30005
Net-IT	INDL	[122]	✓			✓	ALUIF(D)	26	29	17	0
Net-IT	DevopsInfra (network)	[33]	✓	✓			ALH(D)	17	15	25	0
Net-IT	DevopsInfra (product)	[33]	✓				ALCHOQ(D)	22	20	3	2
Net-IT	DevopsInfra (workflow)	[33]				✓	ALCHOQ(D)	13	16	8	2
Net-IT	NORIA-O	[90]	✓	✓	✓	✓	ALCHOI(D)	59	107	71	0
Net-IT	The SEAS Communication ontology	[117]	✓	✓			ALCRIN(D)	69	46	5	6
Net-IT	ToCo	[31]	✓	✓	✓		ALCRI(D)	85	41	54	1
Net-IT	HTTP in RDF	[123]			✓		ALH(D)	15	26	13	0
Process modeling	Activity Ontology	[124]	✓		✓	✓	SRIQ(D)	21	70	12	0
Process modeling	BPMN ontology	[125]			✓	✓	ALCRIQ(D)	158	103	34	0
Process modeling	FOLIO	[126]				✓	ALEQ(D)	29	19	3	0
Process modeling	gist	[127]	✓	✓	✓	✓	SROIQ(D)	138	71	39	15
SE-SI	BOnSAI	[128]	✓			✓	ALCHIN(D)	99	76	41	1
SE-SI	DogOnt	[129]	✓	✓	✓		ALCHOIN(D)	167	18	26	0
SE-SI	IoT-Lite	[130]	✓		✓		ALUI(D)	20	13	9	0
SE-SI	RAMI Ontology	[131]	✓	✓			ALH+ (D)	35	47	19	3
SE-SI	SAREF	[132]	✓	✓	✓	✓	ALCIQ(D)	81	35	5	10
SE-SI	SAREF4SYST	[132]	✓	✓			SRIN	4	9	0	0
SE-SI	SOSA	[133]		✓	✓	✓	ALI(D)	16	21	2	1
SE-SI	The Building Topology Ontology (BOT)	[134]	✓				SRIN	10	16	1	5
SE-SI	mIO! Ontology Network	[135]	✓	✓	✓	✓	SROIQ(D)	624	364	310	520

For models with an implementation, and as per a subset of the evaluation criteria defined in Section 6.1. Abbreviations: $Ref.$ = bibliographical reference, CC = class count, OPC = object property count, DPC = data-type property count, IC = individual count, $SE-SI$ = smart environment & smart industry, $n.a.$ = non applicable.

Table 4: Number of semantic models and facet coverage ratios by application domain.

Theme	MC	St. %	Fu. %	Dy. %	Pr. %	F0 %	F1 %	F2 %	F3 %	F4 %
Generic	18	0,0	11,1	55,6	38,9	**33,3**	33,3	27,8	5,6	0,0
CyberSec	12	50,0	50,00	58,3	83,3	0,0	**41,7**	16,7	0,0	**41,7**
SE-SI	9	**88,9**	**66,7**	55,6	44,4	0,0	11,1	**44,4**	22,2	22,2
Net-IT	8	71,4	42,9	28,6	28,6	0,0	37,5	42,9	14,3	0,0
Process modeling	4	50,0	25,0	**75,0**	**100,0**	0,0	25,0	25,0	**25,0**	25,0
Health Science	1	*100,0*	0,0	0,0	*100,0*	0,0	0,0	*100,0*	0,0	0,0
Overall	52	44,2	36,5	53,8	**55,8**	11,5	**30,8**	**30,8**	9,6	17,3

This table summarizes Table 3 from the perspective of the number of models (MC) and the use of facets by primary application domain ($Theme$). The columns $St.\%$, $Fu.\%$, $Dy.\%$, and $Pr.\%$ correspond to the proportion of models for which the facet has been identified. The columns $Fx\%$ account for the expressiveness of the models by comparing the proportion of models that meet 0, 1, 2, 3, or 4 facets. Italicized values for the Health Science theme indicate that they may not be representative due to the inclusion of only one model from this family in the sample.

However, concepts are missing for a finer grained description of the network topology. Additionally, the ontology mainly focuses on the provisioning activity and is not aligned with other well-known models such as SOSA [133] for sensors and probing systems, and the TM Forum Open API [24] for interoperable definitions of states and operations between DSSs.

The NORIA-O [90] model enables representing network infrastructures and their events – such as alarms and incident tickets – and connects to third-party models for diverse analytical perspectives. However, its generality necessitates careful consideration of how to integrate different technical architectures into a coherent framework for multi-domain analyses. The CRATELO [29] and PACO [137] models enable the representation and classification of network traffic. However, they lack concepts for network topology and operations, which are necessary for contextualizing network traffic sessions within the network topology itself and day-to-day operations. Regarding the representation of *procedural* knowledge, the detailed analysis of the data models also reveals that Semantic Web-based models have less presence in modeling capabilities for knowledge about processes with conditional branching (e.g. IF-THEN-ELSE decision making) compared to sequential and relatedness knowledge. This suggests that representing networks using the available models or through the knowledge graphs formalism may not be self-sufficient for interpreting their dynamics, unless relying on external knowledge and reasoning tools as proposed by FOLIO [126] through the Failure Mode and Effect Analysis (FMEA) approach and the use of a rule engine (e.g. leveraging the Semantic Web Rule Language (SWRL) [138] or the SPARQL Inferencing Notation (SPIN) [139]).

Finally, regarding the compliance with best practices, all models exhibit non-conformance to varying degrees across application domains, in terms of both quantity and severity. However, it is worth noting that the number of non-conformance generally increases with the size of the model.

Table 5 highlights the top five implementation pitfalls found in the set of semantic models listed in Table 3. The relatively low severity of these non-conformance (three "Minor", two "Important"), as well as their nature (i.e. primarily related to the characterization of relationships and the description of concepts and relationships),

suggest that the proposed models are generally ready for short-term integration into a knowledge graph-based solution. However, it is important to note that their inference capabilities may need to be adjusted based on experiments with practical cases.

Table 5: Most common implementation pitfalls in semantic models.

Code	Importance Level	Description	Count
P13	Minor	Inverse relationships not explicitly declared	43
P11	Important	Missing domain or range in properties	42
P08	Minor	Missing annotations	41
P04	Minor	Creating unconnected ontology elements	29
P10	Important	Missing disjointness	20

This table highlights the top five implementation pitfalls found in the set of semantic models listed in Table 3, as reported by the OOPS! tool.

7 Anomaly Detection: a Cartography of Algorithmic Methods

NMSs and SIEMs, described in Section 5.2 as umbrella systems for centralized situation understanding, already enable significant operational efficiency for incident management. However, the remaining limitations discussed in Section 5.3 show room for improvement in their automated analysis functions towards greater explainability, notification contextualization, and notification grouping and prioritizing. With the aim of enabling these improvements, we provide in this section a summary of the anomaly detection methods we found in our systematic literature review, starting with the definition of evaluation criteria in Section 7.1, and reporting on our analysis in Section 7.2.

7.1 Evaluation Criteria

The data collected and evaluation criteria used to analyze the anomaly detection methods are as follows:

- **Name.** The name of the anomaly detection method or, if not available, the title of the document describing it (e.g. research paper, blog post, tool documentation).
- **Primary application domain.** The domain or field of application that originated the anomaly detection method, based on the indications provided by the authors, or if not available, deduced by us.
- **Approach.** A short description of the method, and a categorization of the method summarizing its core principle.
- **Usage step.** The typical stage of the incident management process – as discussed in Section 2.2 and Figure 2 – to which the use of the method corresponds, based on the information provided by the authors, or deduced by us otherwise. We define the following three macro stages: *design* (i.e. techniques providing prior knowledge of system operation and enabling optimal deployment based on safety criteria or

potential re-engineering to meet these criteria), *detection and classification* (i.e. techniques analyzing artifacts related to the system lifecycle to generate an alert for an undesirable situation), and *diagnostic aid* (i.e. techniques enabling the characterization of a given situation).

- **Data structure.** The main data structure(s) used within the algorithmic method for its model learning and inference stages. This data structure may differ from that of the input data due to the specificities of the method (e.g. aggregation of heterogeneous data into a knowledge graph, semantic annotation of natural language documents).

7.2 Algorithmic Methods

In this section, we present the identified algorithmic methods from the literature review, and their categorization based on the previous criteria. Out of 106 references analyzed, 57 emerged with both a primary application domain close to the NetOps and SecOps fields and practicality falling into an incident management stage. We analyze these 57 references in the following paragraphs, providing an overall analysis from the point of view of approaches, usage stage, and data structures. Statistics on the primary application domain are as follows: CyberSec = 22, Net-IT = 17, Generic = 5, Industry 4.0 = 5, Energy systems = 4, Smart-Cities and Smart-Homes (SC-SH) = 2, Business process = 1, and Software engineering = 1. For complete details on these 57 references, including a summary of each approach, we refer to Tables A1 to A4.[27]

Logic for design and diagnosis vs probabilities for detection.

Analyzing the references highlights six families of approaches: **graph-based** where the processing relies on the structure and characteristics of data represented in a graph, with principles drawing from graph theory [140] or message passing [141]; **knowledge-based** where deductive[28] and abductive[29] reasoning leverages domain knowledge organized in taxonomies or ontologies; **Markov model** [142] where a probabilistic model describing the potential state transitions of a system is used for inference; **ML-based** where the probabilistic model used for inference leverages correlations between multiple observational variables as an indicator; **model checking** where a behavioral model in the form of a Finite State Automaton (FSA) [75, 143] is used for inference; **rule-based** where deductive reasoning applies leveraging a set of business rules typically of the IF-THEN-ELSE form. Table 6 shows the distribution of references across these approach families given the three usage stages defined in Section 7.1. The proportions reported in this table indicate a predominance of works applicable to the *detection and classification* stage. Additionally, there is a prevalence of logic-based approaches in the *design* and *diagnostic aid* stages, as opposed to correlation-based approaches in the *detection and classification* stage. These trends

[27] Note that some references discuss multiple approaches, which results in the total number of identified approaches exceeding the number of references.

[28] In an inference process, the formation of a conclusion is based on generally accepted statements or facts (i.e. from general or universal premises).

[29] In an inference process, observational facts (major premise) are evident, but the cause (minor premise) and therefore the conclusion are only probable. Backward chaining on a rule set [51] is a typical implementation of this process, where the system checks if an hypothesis is true or not.

suggest a current focus for anomaly detection on the ability to capture complex situations without necessarily leveraging prior or expert knowledge of the ICT systems. This contrast also suggests a limited current capability for coupling between logic-based and probabilistic approaches. However, the presence of the graph-based approach in all three usage stages suggests that a significant portion of the addressed problems involves the interconnected nature of the data.

Table 6: Approach family and incident management stage in analyzed papers.

Approach	System Design	Detection & Classification	Diagnostic Aid
Rule-based	1	5	0
Model checking	1	3	1
Knowledge-based	**2**	8	**6**
Markov model	0	1	0
Graph-based	1	10	5
ML-based	0	**15**	0
Overall	5 (8,5 %)	**42 (71,2 %)**	12 (20,3 %)

This table provides information on the distribution (in number and proportion) of the analyzed papers, based on the approach family (the middle line serves as an arbitrary separation between logic-based and correlation-based approaches) and the stage of the incident management process involved. Values in bold highlight the most representative approach for a given stage of the incident management process.

Partially ordered sets and graphs as key data structures.

Table 7 shows the distribution of data structures used in algorithmic solutions, as a function of the solution's approach family and usage stage. The following five types of structures emerge from our analysis: data with an **order** relation, which includes timestamped *sequential data* such as event logs and alarms, *network traffic* captures, and *time series* for regularly sampled measurements such as data throughput or temperature; **graph** (*static* or *streaming*) such as network topology; **tabular** data, such as a list of assets with their characteristics; multi-dimensional **data points**; and so-called **mixed** approaches that simultaneously use a combination of the aforementioned structures. From the proportions in Table 7, we observe that data with an order relation are generally predominant across all usage stages. In the *detection and classification* usage stage, approaches primarily utilize data structures – in descending order of preference – such as ordered data, graphs, and tables, with a prevalence of ordered data for ML-based approaches. In the *diagnostic aid* usage stage, approaches make the most use of mixed structures. These observations suggest a general tendency for *detection and classification* approaches to focus on the temporal evolution of systems, while *diagnostic aid* approaches tend to focus on a broader context of the system's state.

Table 7: Data structures within algorithmic methods for anomaly detection.

Approach	Seq. data		Seq. data (network)		Time series		*Ordered (1,2,3)*		Graph		Graph streams		Tabular		Data points		Mixed seq.+ graph		Mixed seq.+ tab.		Mixed seq.+ unstr.		*Mixed (9,10,11)*	
		[%]		[%]		[%]	Σ	[%]		[%]		[%]		[%]		[%]		[%]		[%]		[%]	Σ	[%]
Design																								
G.-based		0,0		0,0		0,0		*0,0*		0,0		0,0		0,0		0,0	1	10,0		0,0		0,0	*1*	*7,1*
K.-based		0,0		0,0		0,0		*0,0*		0,0		0,0		0,0		0,0	1	10,0	1	**33,3**		0,0	*2*	*14,3*
M. check.	1	6,7		0,0		0,0	*1*	*3,7*		0,0		0,0		0,0		0,0		0,0		0,0		0,0		*0,0*
R.-based		0,0		0,0		0,0		*0,0*	1	9,1		0,0		0,0		0,0		0,0		0,0		0,0		*0,0*
Detection & Classification																								
G.-based	2	13,3		0,0	1	16,7	*3*	*11,1*	3	**27,3**	1	**50,0**	2	**66,7**		0,0	1	10,0		0,0		0,0	*1*	*7,1*
K.-based	3	20,0	1	16,7		0,0	*4*	*14,8*	3	**27,3**		0,0		0,0		0,0		0,0	1	**33,3**		0,0	*1*	*7,1*
Markov	1	6,7		0,0		0,0	*1*	*3,7*		0,0		0,0		0,0		0,0		0,0		0,0		0,0		*0,0*
ML-based	5	**33,3**	1	16,7	5	**83,3**	*11*	***40,7***		0,0	1	**50,0**		0,0	2	**100,0**		0,0	1	**33,3**		0,0	1	*7,1*
M. check.	1	6,7	1	16,7		0,0	*2*	*7,4*	1	9,1		0,0		0,0		0,0		0,0		0,0		0,0		*0,0*
R.-based	1	6,7	3	**50,0**		0,0	*4*	*14,8*	1	9,1		0,0		0,0		0,0		0,0		0,0		0,0		*0,0*
Diagnostic Aid																								
G.-based		0,0		0,0		0,0		*0,0*		0,0		0,0		0,0		0,0	5	**50,0**		0,0		0,0	*5*	***35,7***
K.-based		0,0		0,0		0,0		*0,0*	2	18,2		0,0	1	33,3		0,0	2	20,0		0,0	1	**100,0**	*3*	*21,4*
M. check.	1	6,7		0,0		0,0	*1*	*3,7*		0,0		0,0		0,0		0,0		0,0		0,0		0,0		*0,0*
Overall	15	**25,4**	6	10,2	6	10,2	*27*	***45,8***	11	18,6	2	3,4	3	5,1	2	3,4	10	16,9	3	5,1	1	1,7	*14*	*23,7*

This table provides information on the distribution (in number and proportion) of the main data structures used within the algorithmic solutions in the analyzed papers, based on the algorithmic approach family and the stage of the incident management process involved. Values in bold highlight the most representative approach for a given data structure. The columns in italics represent cumulative values (*ordered* = columns 1 + 2 + 3, *mixed* = columns 9 + 10 + 11) to provide a summary view of similar structures.

8 Conclusion

With the aim of facilitating the capture and interpretation of complex situations occurring on networks, we have proposed a review of NMS and SIEM DSSs capabilities, semantic models, and algorithmic solutions that, through their individual improvement or combination, could allow to achieve both crisp and approximate reasoning on the interplay between a large-scale ICT system architecture and its operation. Table 8 summarizes our key findings. It highlights that NMS and SIEM DSSs are well-established tools that simplify the analysis of diverse data sources (e.g. assets database, logs and alarms from IPoDWDM and VM management systems, vulnerability scans, etc.). However, their effectiveness is hindered by the implicit heterogeneity of these sources, which limits the ability to contextualize network and service failures and implement comprehensive analysis solutions that consider a broader range of information, including network topology. Furthermore, while there are semantic models that align with the discourse domain of NetOps and SecOps, enabling the utilization of knowledge graphs for knowledge representation, they do not individually – except for a few exceptions – fully cover the necessary discourse domain. Additionally, they do not inherently support reasoning about system state changes related to procedures with conditional branching in decision-making processes. Finally, various algorithmic methods exist to address key steps in the incident management process. However, individually, they do not capture and analyze phenomena that involve temporal, structural, logical, and probabilistic aspects simultaneously.

Advancing through knowledge engineering and massive data integration.

Data heterogeneity and interrelatedness between data entities (i.e. distinct and persistent units of information) appear to be cornerstones for advancing the capabilities of DSSs. In this survey, we assumed that knowledge graphs naturally align with these two notions in the sense that they bring an abstraction level for standard interpretation and logical reasoning over heterogeneous data. Sketching a next-generation NMS/SIEM therefore leads to understand how to bring knowledge graphs to such a system, while considering that NMSs and SIEMs systems rely on a multitude of both streamed and static data sources.

Several tools have been proposed in different application domains for Knowledge Graph Construction (KGC). For streamed data: RMLStreamer [144] applies declarative mapping on the fly to structured data streams (e.g. file, Kafka topic) with RDF Mapping Language (RML) [145] rules; StreamingMASSIF [146] uses basic string substitution for mapping, and allows for real-time reasoning (e.g. SPARQL query processing, Complex Event Time processing); C-SPARQL [147] extends the SPARQL query language for continuous reasoning within a publisher/subscriber platform. For static data: RMLMapper [145] enables data fetching and declarative mapping with RML rules; Ontop [148] creates a virtual graph representation of various data sources via SPARQL queries; and SLOGERT [101] orchestrates log modeling and annotation with Cybersecurity Threat Intelligence (CTI) tags[30].

[30] As for SLOGERT v0.9.1: with MITRE CEE categories from http://cee.mitre.org/language/1.0-alpha/

Table 8: Key findings from the survey.

KR	CX	AD	XP
Decision Support System – Section 5			
• Data is generally recorded in a structure close to its original format, typically in a tabular structure, without annotations. • Data normalization can be performed at different stages of the data ingestion process, depending on the design choices of the DSS.	• The data recording/management block utilizes a combination of storage technologies and often follows a hub/aggregator architecture.	• The analysis block can utilize internal resources and/or external services. • Correlation-based analysis on top of rule-based characterization is common. • Automated contextualization may not consider network topology or network operation information.	• Explainability is addressed by contextualizing notifications through rendering, rewriting, and enriching.
Semantic models – Section 6			
• There are many models with good overall quality, but none fully cover all aspects required for reasoning on the interplay between network architecture and its operation. • Some models describe network topology but at a high level or for specific domains.	• Current data models reveal various opportunities for enrichment, improvement, or linking of the models. • Inference capabilities may need to be adjusted based on experiments with practical cases.	• Anomaly detection with semantic models currently generally rely on graph traversal or rule-based techniques. • Semantic Web-based models have limited capabilities for conditional branching knowledge.	• Explainability is naturally addressed for semantic models due to their explicit knowledge representation, roots in logic, and utilization of shared vocabularies.
Algorithmic methods – Section 7			
• Data with an order relation is generally predominant across all usage stages. • In the detection and classification stage, approaches primarily use ordered data, graphs, and tables, with a prevalence of ordered data for machine learning-based approaches. • Detection and classification approaches focus on the temporal evolution of systems, while diagnostic aid approaches consider a broader context of the system's state (typically using mixed structures).	• There are few solutions directly designed for streamed analysis, and they generally require a training phase.	• Focus on detection and classification in current works on anomaly detection. • Focus on the capability to capture complex situations without relying heavily on prior or expert knowledge of ICT systems in current works on anomaly detection. • Interconnected data (e.g. graphs) is important, but its usage is not widespread.	• Explainability is naturally addressed for algorithmic methods with roots in logic. • Limited current capability for coupling between logic-based and probabilistic approaches.

Summary of the key messages from the *capabilities of DSSs*, *semantic models*, and *algorithmic methods for anomaly detection* aspects according to the research axes Knowledge Representation (KR), CompleXity (CX), Anomaly Detection (AD), and eXPlainability (XP) defined in Section 4.

These solutions provide a foundation towards a Knowledge Graph-based NMS/SIEM. However, additional research effort is required to achieve an end-to-end solution design that bridges the Semantic Web tools with the design patterns observed in industrial DSSs. It includes satisfying requirements for distributed processing, separation of concerns, data sketching (i.e. enabling both early and posterior reasoning on data), openness to third-party databases/tools, and re-use of well-established frameworks (e.g. declarative data transformation, message passing). Indeed, in end-to-end frameworks [149–152], the KGC step is never considered singular, initial or terminal, but rather is the subject of multiple instances of a similar tool/principle within processing flows depending on the application field. In addition, this step is always placed between heterogeneous non-RDF data and a knowledge graph working sometimes as a main data storage, and sometimes as a support for third-party inference processes. This variety of options in itself constitutes a field of exploration to be pursued.

Regarding the standardized interpretation of data, the variety of available semantic models encourages understanding how to leverage existing implementations of vocabularies without introducing complexity. This can be achieved by avoiding using vocabularies with loosely coupled semantics to the application domain, avoiding the introduction of a new ontology that lacks interoperability with other standard vocabularies, and preventing the creation of an ontology network that would unnecessarily lengthen reasoning paths. Various knowledge engineering methodologies allow for approaching this research, whether it be general approaches such as practical guides to semantic modelling [153] and ontology design patterns [154], or more focused approaches: Competency Questions [155], Dichoscope [156], DOE [157], NeOn [158], OntoClean [159], ontology design with Formal Concept Analysis (FCA) [160], automated ontology learning from raw data with Text2Onto [161], automated derivation of class taxonomies from an already existing knowledge graph [162], and translation of formal models to an ontology [163–165].

Advancing in anomaly detection and explainability.

Put simply, anomaly detection methods in the context of NetOps and SecOps aim to identify deviations from normal behavior and flag potentially undesirable/suspicious activities at both the ICT system and user level. While various approaches have been proposed, as discussed in Section 7, only a few of them simultaneously combine intrinsic explainability through the use of explicit representations (e.g. knowledge graphs, FSAs, Petri nets) and the ability to handle interconnected multi-dimensional data, including the temporal dimension. Assuming the use of knowledge graphs as a foundational formalism for NMS/SIEM DSSs, this prompts us to understand to what extent the existing algorithmic solutions are suited to it.

Focusing on activity modeling and analysis, trace-based reasoning [166] shows opportunities in creating tools for semantically interpreting digital services artifacts using controlled vocabularies and semantic models. Indeed, the representation of events and activities within knowledge graphs is implemented across a range of data models, encompassing both domain-independent and domain-specific contexts: process modeling and execution (BBO [167], Petri nets-related [168, 169], HTTPinRDF [123, 170]); causal analysis (FARO [108]); cyber-security and network

operations (UCO [30], MITRE D3FEND [92], NORIA-O [90]); smart cities (iCity ActivityOntology [171]).

At the same time, User and Entity Behavior Analytics (UEBA) corresponds to the temporal dimension of the knowledge graph, which further motivates the search for how an anomaly context capture (e.g. a subgraph centered around the undesirable event) [172] can accommodate historical graphs [173] or a graph that evolves over time. Graph embeddings [174] typically allow for capturing the context of graph entities. However, since learning embeddings (i.e. a vector representation of the subgraph that enables similarity calculations) is a potentially time-consuming process performed on a snapshot of the graph, using such an approach requires defining the content of the subgraphs to capture [175], particularly in relation to the speed of graph evolution. Moreover, since such an approach may also not be generalizable, the simultaneous implementation of various approaches can prove useful, whether by following the principles of cooperative decision making [41] (i.e. a heuristic approach for problem-solving by using results obtained from complementary inference techniques) or by combining properties of inference models into a single one [176–179] (e.g. by including the power of logical reasoning into ML-based models).

References

[1] ITU-T: X.200 : Information Technology - Open Systems Interconnection - Basic Reference Model: The Basic Model. Recommendation X.200 (07/94), International Telecommunication Union (ITU) (1994)

[2] ITU-T: G.7710: Common Equipment Management Function Requirements. Recommendation G.7710/Y.1701, International Telecommunication Union (ITU) (2020)

[3] ITU-T/CCITT: ITU-T Rec. X.733 (02/92) Information technology - Open Systems Interconnection - Systems Management: Alarm reporting function. Recommendation, International Telecommunication Union (ITU) (1992)

[4] CNSS Glossary Working Group: CNSSI 4009. Technical Report CNSSI No. 4009, Committee on National Security Systems (CNSS) (2015)

[5] ETSI: Method and pro Forma for Threat, Vulnerability, Risk Analysis (TVRA). Technical Specification ETSI TS 102 165-1 V5.2.3 (2017-10), ETSI (2017)

[6] ISO/IEC JTC 1/SC 40: Information technology – Service management – Part 1: Service management system requirements. Technical Report 20000-1:2018, International Organization for Standardization/International Electrotechnical Commission (2018)

[7] Alison Cartlidge, Ashley Hanna, Colin Rudd, Ivor Macfarlane, John Windebank, Stuart Rance: An Introductory Overview of ITIL V3. The UK Chapter of the itSMF, Bracknell, UK (2007)

[8] ITEMO: Standards for lightweight IT service management (FitSM). https://www.fitsm.eu/ (2022)

[9] Joint Task Force: Security and Privacy Controls for Information Systems and Organizations. Technical Report NIST SP 800-53r5, National Institute of Standards and Technology (2020). https://doi.org/10.6028/NIST.SP.800-53r5

[10] Richard Caralli, James F. Stevens, Lisa R. Young, William R. Wilson: Introducing OCTAVE Allegro: Improving the Information Security Risk Assessment Process. Technical report, Carnegie Mellon University (2007). https://doi.org/10.1184/R1/6574790.V1

[11] ANSSI: EBIOS Risk Manager. Technical Report ANSSI-PA-048, Agence Nationale de la Sécurité des Systèmes d'Information (ANSSI) (2018). https://cyber.gouv.fr/sites/default/files/2018/10/guide-methode-ebios-risk-manager.pdf

[12] Paul Cichonski, Tom Millar, Tim Grance, Karen Scarfone: Computer Security Incident Handling Guide: Recommendations of the National Institute of Standards and Technology. Technical Report NIST SP 800-61r2, National Institute of Standards and Technology (2012). https://doi.org/10.6028/NIST.SP.800-61r2

[13] Quentin Rousseau: Incident Management vs. Incident Response – What's the Difference? https://rootly.com/blog/incident-management-vs-incident-response-what-s-the-difference (2021)

[14] Stefan Kempter: IT Process Maps – Incident Management. https://wiki.en.it-processmaps.com/index.php/Incident_Management (2007)

[15] ISO/IEC: Information security, cybersecurity and privacy protection – Information security management systems – Requirements. Technical Report 27001:2022, ISO/IEC (2022)

[16] International Atomic Energy Agency: Design Basis Threat (DBT). IAEA (2019)

[17] David Swift: A Practical Application of SIM/SEM/SIEM Automating Threat Identification. White Paper, SANS Institute (2007)

[18] Dauphin-Tanguy, G.: Les Bond Graphs et Leur Application En Mécatronique. Techniques de l'Ingénieur (1999)

[19] Wang, X., Song, J., Zhou, X.: Hypergraph Based Network Model and Architechture for Deep Space Exploration. In: Information Computing and Applications. Springer Berlin Heidelberg, Berlin, Heidelberg (2011)

[20] Domke, J., Hoefler, T., Matsuoka, S.: Fail-in-Place Network Design: Interaction Between Topology, Routing Algorithm and Failures. In: SC '14: Proceedings

of the International Conference for High Performance Computing, Networking, Storage and Analysis (2014). https://doi.org/10.1109/SC.2014.54

[21] Manish Thapa, Jose Espejo-Uribe, Evangelos Pournaras: Measuring Network Reliability and Repairability against Cascading Failures. Journal of Intelligent Information Systems (2019) https://doi.org/10.1007/s10844-017-0477-0

[22] Yechiel Levin, Batami Gold, Dan Mabee, Atik Mapari, Kent Sharkey, Robert Lyon, David Coulter: Advanced Threat Detection with User and Entity Behavior Analytics (UEBA) in Microsoft Sentinel. https://learn.microsoft.com/en-us/azure/sentinel/identify-threats-with-entity-behavior-analytics (2024)

[23] Fedor, M., Schoffstall, M.L., Davin, J.R., Case, D.J.D.: Simple Network Management Protocol (SNMP). RFC Editor (1990). https://doi.org/10.17487/RFC1157

[24] TM Forum: TM Forum Open APIs. https://github.com/tmforum-apis (2018)

[25] Open Networking Foundation: YANG, OpenConfig, and gNMI. https://opennetworking.org/wp-content/uploads/2019/10/NG-SDN-Tutorial-Session-2.pdf (2019)

[26] Aidan Hogan, Eva Blomqvist, Michael Cochez, Claudia d'Amato, Gerard de Melo, Claudio Gutierrez, José Emilio Labra Gayo, Sabrina Kirrane, Sebastian Neumaier, Axel Polleres, Navigli, R., Axel-Cyrille Ngonga Ngomo, Sabbir M. Rashid, Anisa Rula, Lukas Schmelzeisen, Juan Sequeda, Steffen Staab, Antoine Zimmermann: Knowledge Graphs (2020)

[27] Tim Berners-Lee, James Hendler, Ora Lassila: The Semantic Web – A New Form of Web Content That Is Meaningful to Computers Will Unleash a Revolution of New Possibilities. Scientific American (2001)

[28] Mattijs Ghijsen, Jeroen Van Der Ham, Paola Grosso, Cosmin Dumitru, Hao Zhu, Zhiming Zhao, Cees De Laat: A Semantic-Web Approach for Modeling Computing Infrastructures. Computers & Electrical Engineering (2013) https://doi.org/10.1016/j.compeleceng.2013.08.011

[29] Alessandro Oltramari, Loria Cranor, Robert Walls, Patrick McDaniel: Building an Ontology of Cyber Security. In: 9^{th} Conference on Semantic Technologies for Intelligence, Defense, and Security (STIDS) (2014)

[30] Zareen Syed, Ankur Padia, M. Lisa Mathews, Tim Finin, Anupam Joshi: UCO: A Unified Cybersecurity Ontology. In: AAAI Workshop on Artificial Intelligence for Cyber Security. AAAI Press, Washington DC, USA (2016)

[31] Qianru Zhou, Alasdair J. G. Gray, Stephen McLaughlin: ToCo: An Ontology for

Representing Hybrid Telecommunication Networks. In: 16^{th} European Semantic Web Conference (ESWC). Springer, Cham (2019). https://doi.org/10.1007/978-3-030-21348-0_33

[32] Andrei Brazhuk: Threat Modeling of Cloud Systems with Ontological Security Pattern Catalog. International Journal of Open Information Technologies (2021)

[33] Oscar Corcho, David Chaves-Fraga, Jhon Toledo, Julián Arenas-Guerrero, Carlos Badenes-Olmedo, Mingxue Wang, Hu Peng, Nicholas Burrett, José Mora, Puchao Zhang: A High-Level Ontology Network for ICT Infrastructures. In: 20^{th} International Semantic Web Conference (ISWC) (2021). https://doi.org/10.1007/978-3-030-88361-4_26

[34] Yassine Naghmouchi, Nancy Perrot, Nizar Kheir, Ali Ridha Mahjoub, Jean-Philippe Wary: A New Risk Assessment Framework Using Graph Theory for Complex ICT Systems. In: Proceedings of the 8^{th} ACM CCS International Workshop on Managing Insider Security Threats (2016). https://doi.org/10.1145/2995959.2995969

[35] Marwan Ghanem, Clémence Magnien, Fabien Tarissan: How to Exploit Structural Properties of Dynamic Networks to Detect Nodes with High Temporal Closeness. In: Cologne-Twente Workshop on Graphs and Combinatorial Optimization 2018 (CTW'18), Paris, France (2018)

[36] Lucas Potin, Figueiredo, R., Vincent Labatut, Christine Largeron: Pattern Mining for Anomaly Detection in Graphs: Application to Fraud in Public Procurement. In: Machine Learning and Knowledge Discovery in Databases: Applied Data Science and Demo Track. Springer, Cham (2023). https://doi.org/10.1007/978-3-031-43427-3_5

[37] Andrew Clark, George Mohay, Bradley Schatz: Rich Event Representation for Computer Forensics. In: Proceedings of the Fifth Asia Pacific Industrial Engineering and Management Systems Conference, Gold Coast, Australia (2004)

[38] Maria Massri: Designing a Temporal Graph Management System for IoT Application Domains. PhD thesis, Université Rennes 1, Rennes, France (2022)

[39] VirusTotal: YARA. https://github.com/VirusTotal/yara (2021)

[40] Iman Akbari, Mohammad A. Salahuddin, Leni Ven, Noura Limam, Raouf Boutaba, Bertrand Mathieu, Stephanie Moteau, Stephane Tuffin: Traffic Classification in an Increasingly Encrypted Web. Communications of the ACM (2022)

[41] Ben Goertzel, Cassio Pennachin, Nil Geisweiller: Engineering General Intelligence, Part 1: A Path to Advanced AGI Via Embodied Learning and Cognitive

Synergy. Atlantis Press, The Netherlands (2014). https://doi.org/10.2991/978-94-6239-027-0

[42] Pankaj Malhotra, Lovekesh Vig, Gautam Shroff, Puneet Agarwal: Long Short Term Memory Networks for Anomaly Detection in Time Series. In: Computational Intelligence and Machine Learning, Bruges, Belgium (2015)

[43] Anouk Barberousse: Observation et Calcul – Les Données Traitées Informatiquement Sont-Elles Encore Des Données d'observation ? (2011)

[44] Costa, M., Crowcroft, J., Castro, M., Rowstron, A., Zhou, L., Zhang, L., Barham, P., Rowstron, A.: Vigilante: End-to-End Containment of Internet Worm Epidemics. ACM Transactions on Computer Systems (2008) https://doi.org/10.1145/1455258.1455259

[45] Randall Davis: Reasoning from First Principles in Electronic Troubleshooting. International Journal of Man-Machine Studies (1983) https://doi.org/10.1016/S0020-7373(83)80063-7

[46] Eyke Hüllermeier: Case-Based Approximate Reasoning. Theory and Decision Library B. Springer, Cham (2007). https://doi.org/10.1007/1-4020-5695-8

[47] Dan Geiger, Judea Pearl: Logical and Algorithmic Properties of Independence and Their Application to Bayesian Networks. Annals of Mathematics and Artificial Intelligence (1990) https://doi.org/10.1007/BF01531004

[48] Bram Steenwinckel: IBCNServices/Folio-Ontology. https://github.com/IBCNServices/Folio-Ontology (2019)

[49] Andreas M. Hein, Stephen Baxter: Artificial Intelligence for Interstellar Travel (2018)

[50] W3C SPARQL Working Group: SPARQL Protocol and RDF Query Language 1.1 (SPARQL). W3C Recommendation, W3C (2013)

[51] Mark Proctor: Drools: A Rule Engine for Complex Event Processing. In: Applications of Graph Transformations with Industrial Relevance. Springer, Cham (2012). https://doi.org/10.1007/978-3-642-34176-2_2

[52] Franz Baader, Deborah L. McGuinness, Daniele Nardi, Peter F. Patel-Schneider: The Description Logic Handbook: Theory, Implementation and Applications. Cambridge University Press, UK (2003)

[53] Miguel Hernan, Jamie Robins: Causal Inference: What If. Harvard College, Cambridge, USA (2020). https://www.hsph.harvard.edu/miguel-hernan/causal-inference-book/

[54] Erkuden Rios, Eider Iturbe, Angel Rego, Nicolas Ferry, Jean-Yves Tigli,

Stéphane Lavirotte, Gerald Rocher, Phu Nguyen, Hui Song, Rustem Dautov, Wissam Mallouli, Ana Rosa Cavalli: The DYNABIC Approach to Resilience of Critical Infrastructures. In: Proceedings of the 18^{th} International Conference on Availability, Reliability and Security. ARES '23. Association for Computing Machinery, ??? (2023). https://doi.org/10.1145/3600160.3605055

[55] David Allison, Paul Smith, Kieran Mclaughlin: Digital Twin-Enhanced Incident Response for Cyber-Physical Systems. In: Proceedings of the 18^{th} International Conference on Availability, Reliability and Security. ARES '23. Association for Computing Machinery, New York, NY, USA (2023). https://doi.org/10.1145/3600160.3600195

[56] Lonvick, C.M., Ylonen, T.: The Secure Shell (SSH) Connection Protocol. RFC Editor (2006). https://doi.org/10.17487/RFC4254

[57] Claise, B.: Cisco Systems NetFlow Services Export Version 9. RFC Editor (2004). https://doi.org/10.17487/RFC3954

[58] Pankaj Prasad, Josh Chessman: Market Guide for IT Infrastructure Monitoring Tools. Technical Report G00450400, Gartner (2019)

[59] Josh Chessman: Magic Quadrant for Network Performance Monitoring and Diagnostics. Technical Report G00463582, Gartner (2020)

[60] Kelly Kavanagh, Toby Bussa, Gorka Sadowski: Magic Quadrant for Security Information and Event Management. Technical Report G00348811, Gartner (2018)

[61] Gustavo González-Granadillo, Susana González-Zarzosa, Rodrigo Diaz: Security Information and Event Management (SIEM): Analysis, Trends, and Usage in Critical Infrastructures. Sensors (2021) https://doi.org/10.3390/s21144759

[62] HappyFox Inc.: HappyFox Help Desk. https://www.happyfox.com/

[63] Houghton, B.K.: Terrorism Knowledge Base: A Eulogy (2004-2008). Perspectives on terrorism (2010)

[64] CISCO Systems: Snort – Network Intrusion Detection & Prevention System. https://www.snort.org/

[65] The Zeek Project: The Zeek Network Security Monitor. https://zeek.org/. Formerly Bro IDS

[66] Rob Kitchin, Gavin McArdle: What Makes Big Data, Big Data? Exploring the Ontological Characteristics of 26 Datasets. Big Data & Society (2016) https://doi.org/10.1177/2053951716631130

[67] Zenoss: ZenOSS Logical Model. http://wiki.zenoss.org/ZenOSS_Logical_Model

(2014)

[68] RSA-IDD-Legacy: Global NWP 11.5 Architecture Diagram. RSA (2021)

[69] Eric Freeman, Elisabeth Robson, Kathy Sierra, Bert Bates (eds.): Head First Design Patterns. O'Reilly, Sebastopol, CA (2004)

[70] Jean-François Nogier, Thierry Bouillot, Jules Leclerc: Ergonomie Des Interfaces: Guide Pratique Pour La Conception Des Applications Web, Logicielles, Mobiles et Tactiles. Dunod, Paris (2011)

[71] NIST: National Vulnerability Database (NVD). https://nvd.nist.gov/

[72] MISP project: MISP – Open Source Threat Intelligence Platform & Open Standards For Threat Information Sharing. https://www.misp-project.org/. Formely known as Malware Information Sharing Platform

[73] Technologies, M.: Maltego. https://www.maltego.com/

[74] OASIS Open: Introduction to STIX – Structured Threat Information eXpression. https://oasis-open.github.io/cti-documentation/stix/intro

[75] Gilbert Laycoc: The Theory and Practice of Specification Based Software Testing. PhD thesis, University of Sheffield, Department of Computer Science (1992)

[76] Kent Erickson: Layer 2 Network Connection Awareness Improves Root Cause Analysis. https://www.zenoss.com/blog/layer-2-network-connection-awareness-improves-root-cause-analysis (2015)

[77] Riverbed Technology: Application Performance Monitoring for Microservices-Based Applications. https://www.aternity.com/blogs/monitoring-a-microservices-based-application/ (2017)

[78] Rob Bolton: Anomaly Detection and Root Cause Analysis with AppDynamics Cognition Engine. https://www.appdynamics.com/blog/product/anomaly-detection-root-cause-analysis-appdynamics-cognition-engine/ (2019)

[79] IBM: IBM Security I2 Enterprise Insight Analysis – Overview. https://www.ibm.com/products/i2-enterprise-insight-analysis (2020)

[80] IEEE: IEEE Standard for Ethernet. Technical report, IEEE (2018). https://doi.org/10.1109/IEEESTD.2018.8457469

[81] Björklund, M.: The YANG 1.1 Data Modeling Language. RFC Editor (2016). https://doi.org/10.17487/RFC7950 . https://www.rfc-editor.org/info/rfc7950

[82] Zseby, T., Claise, B., Quittek, J., Zander, S.: Requirements for IP Flow Information Export (IPFIX). RFC Editor (2004). https://doi.org/10.17487/RFC3917 . https://www.rfc-editor.org/info/rfc3917

[83] European Commission: Directive on measures for a high common level of cybersecurity across the Union (NIS2 Directive). http://data.europa.eu/eli/dir/2022/2555/2022-12-27 (2022)

[84] Fovino, I.N., Neisse, R., Lazari, A., Ruzzante, G., Polemi, N., Figwer, M.: European Cybersecurity Centres of Expertise Map: Definitions and Taxonomy. Technical Report JRC 111441, European Union (2018)

[85] The MITRE Corporation: Cyber Security Measurement & Management Architecture. https://measurablesecurity.mitre.org/docs/Cyber_Security_Measurement_and_Management_Poster.pdf (2013)

[86] Norbert Wiener, Ronan Le Roux, Robert Vallée, Nicole Vallée: La Cybernétique: Information et Régulation Dans Le Vivant et La Machine. Sources Du Savoir. Éd. du Seuil, Paris (2014)

[87] Gustave Guillaume, Olivier Soutet, Roch Valin: Temps et Verbe Suivi de L'architectonique du Temps dans les Langues Classiques: Théorie des Aspects, des Modes et des Temps, Reproduction en fac-similé edn. Honoré Champion éditeur, Paris (2021)

[88] Mark A. Musen: The Protégé Project: A Look Back and a Look Forward. AI matters (2015) https://doi.org/10.1145/2757001.2757003

[89] María Poveda-Villalón, Gómez-Pérez, A., Mari Carmen Suárez-Figueroa: OOPS! (OntOlogy Pitfall Scanner!): An on-Line Tool for Ontology Evaluation. International Journal on Semantic Web and Information Systems (IJSWIS) (2014)

[90] Tailhardat, L., Chabot, Y., Troncy, R.: NORIA-O: an Ontology for Anomaly Detection and Incident Management in ICT Systems. In: Semantic Web – 21^{st} International Conference, ESWC 2024, Hersonissos, Crete, Greece, May 26 - 30, 2024, Proceedings (2024)

[91] Ismail Harrando, Raphaël Troncy: Explainable Zero-Shot Topic Extraction Using a Common-Sense Knowledge Graph. In: 3^{rd} Conference on Language, Data and Knowledge (LDK 2021). Open Access Series in Informatics (OASIcs). Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl, Germany (2021). https://doi.org/10.4230/OASIcs.LDK.2021.17

[92] Peter E. Kaloroumakis, Michael J. Smith: Toward a Knowledge Graph of Cybersecurity Countermeasures. Technical report, The MITRE Corporation (2021)

[93] Richard Cyganiak, David Wood, Markus Lanthaler: RDF 1.1 Concepts and Abstract Syntax. W3C Recommendation, W3C (2014)

[94] Boris Motik, Peter F. Patel-Schneider, Bijan Parsia, Conrad Bock, Achille Fokoue, Peter Haase, Rinke Hoekstra, Ian Horrocks, Alan Ruttenberg, Uli Sattler, Michael Smith: OWL 2 Web Ontology Language – Structural Specification and Functional-Style Syntax (Second Edition). W3C Recommendation, W3C (2012)

[95] Eoghan Casey, Sean Barnum, Ryan Griffith, Jonathan Snyder, Harm van Beek, Alex Nelson: Advancing Coordinated Cyber-Investigations and Tool Interoperability Using a Community Developed Specification Language. Digital Investigation (2017) https://doi.org/10.1016/j.diin.2017.08.002

[96] Malek Ben Salem, Chris Wacek: Enabling New Technologies for Cyber Security Defense with the ICAS Cyber Security Ontology. In: Proceedings of the Tenth Conference on Semantic Technology for Intelligence, Defense, and Security, Fairfax VA, USA (2015)

[97] Kabul Kurniawan, Elmar Kiesling, Dietmar Winkler, Andreas Ekelhart: The ICS-SEC KG: An Integrated Cybersecurity Resource for Industrial Control Systems. In: The Semantic Web — ISWC 2024. Springer. https://doi.org/10.1007/978-3-031-77847-6_9

[98] Nidhi Rastogi, Sharmishtha Dutta, Mohammed J. Zaki, Alex Gittens, Charu Aggarwal: MALOnt: An Ontology for Malware Threat Intelligence (2020). https://doi.org/10.13140/RG.2.2.16426.64962

[99] Yoan Chabot: Construction, enrichment and semantic analysis of timelines : Application to digital forensics. PhD thesis, University of Burgundy (2015)

[100] Andrei Brazhuk: Security Patterns Based Approach to Automatically Select Mitigations in Ontology-Driven Threat Modelling. In: Open Semantic Technologies for Intelligent Systems (OSTIS) (2020)

[101] Andreas Ekelhart, Fajar J. Ekaputra, Elmar Kiesling: The SLOGERT Framework for Automated Log Knowledge Graph Construction. In: The Semantic Web. Springer, Cham (2021). https://doi.org/10.1007/978-3-030-77385-4_38

[102] Faranak Sobhani, Umberto Straccia: Towards a Forensic Event Ontology to Assist Video Surveillance-based Vandalism Detection. In: Proceedings of the 34^{th} Italian Conference on Computational Logic, Trieste, Italy (2019)

[103] Krisnadhi, A., Hitzler, P.: A Core Pattern for Events. In: Advances in Ontology Design and Patterns [revised and Extended Versions of the Papers Presented at the 7^{th} Edition of the Workshop on Ontology and Semantic Web Patterns, WOP@ISWC 2016, Kobe, Japan, 18^{th} October 2016]. Studies on the Semantic Web. IOS Press, Amsterdam, The Netherlands (2016). https://doi.org/10.3233/

978-1-61499-826-6-29

[104] William R. Hogan: Geographical Entity Ontology. https://github.com/ufbmi/geographical-entity-ontology/wiki (2016)

[105] Franck Berthelon, Peter Sander: Emotion Ontology for Context Awareness. In: 2013 IEEE 4^{th} International Conference on Cognitive Infocommunications (CogInfoCom) (2013). https://doi.org/10.1109/CogInfoCom.2013.6719313

[106] Yves Raimond, Samer Abdallah: The Event Ontology. http://motools.sourceforge.net/event/event.html (2007)

[107] Simon Gottschalk, Elena Demidova: EventKG: A Multilingual Event-Centric Temporal Knowledge Graph. In: The Semantic Web. Springer, ??? (2018). https://doi.org/10.1007/978-3-319-93417-4_18

[108] Youssra Rebboud, Pasquale Lisena, Raphael Troncy: Beyond Causality: Representing Event Relations in Knowledge Graphs. In: Knowledge Engineering and Knowledge Management. Springer International, Cham (2022)

[109] Dan Brickley, Libby Miller: Friend of a Friend (FOAF) Vocabulary Specification . http://xmlns.com/foaf/spec/ (2004)

[110] Nicholas J. Car, Timo Homburg, Matthew Perry, John Herring, Frans Knibbe, Simon J.D. Cox, Joseph Abhayaratna, Mathias Bonduel: OGC GeoSPARQL – A Geographic Query Language for RDF Data. OGC Implementation Standard, Open Geospatial Consortium (2022)

[111] Harry Chen, Dan Brickley: Geonames ontology in OWL. https://web.archive.org/web/20080212144050/http://www.geospatialsemanticweb.com/2006/10/14/geonames-ontology-in-owl (2006)

[112] Claudia Villalonga, Muhammad Razzaq, Wajahat Khan, Hector Pomares, Ignacio Rojas, Sungyoung Lee, Oresti Banos: Ontology-Based High-Level Context Inference for Human Behavior Identification. Sensors (2016) https://doi.org/10.3390/s16101617

[113] Hajo Rijgersberg, Mark van Assem, Jan Top: Ontology of Units of Measure and Related Concepts. Semantic Web (2013) https://doi.org/10.3233/SW-2012-0069

[114] David Martin, Mark Burstein, Jerry Hobbs, Ora Lassila, Drew McDermott, Sheila McIlraith, Srini Narayanan, Massimo Paolucci, Bijan Parsia, Terry Payne, Evren Sirin, Naveen Srinivasan, Katia Sycara: OWL-S: Semantic Markup for Web Services. Technical Report OWL-S 1.1, DARPA Agent Markup Language (DAML) Program (2004)

[115] Simon Cox, Chris Little: Time Ontology in OWL. Candidate Recommendation OGC 16-071r3, W3C (2020)

[116] Timothy Lebo, Satya Sahoo, Deborah McGuinness: PROV-O: The PROV Ontology. W3C Recommendation TR/prov-o, W3C (2013)

[117] Maxime Lefrançois, Jarmo Kalaoja, Takoua Ghariani, Antoine Zimmermann: SEAS Knowledge Model. Deliverable 2.2, ITEA2 12004 Smart Energy Aware Systems (2016)

[118] Ralph Hodgson: QUDT: Quantities, Units, Dimensions and Data Types Ontologies. QUDT.org (2011). https://doi.org/10.25504/FAIRSHARING.D3PQW7

[119] Armin Haller, Krzysztof Janowicz, Simon Cox, Danh Le Phuoc, Kerry Taylor, Maxime Lefrançois: Semantic Sensor Network Ontology. W3C Recommendation OGC 16-079, W3C (2017)

[120] Dan Brickley, Leigh Dodds, Libby Miller: Term-Centric Semantic Web Vocabulary Annotations. W3c interest group note, W3C (2009)

[121] Mauro Dragoni, Tania Bailoni, Rosa Maimone, Claudio Eccher: HeLiS: An Ontology for Supporting Healthy Lifestyles. In: The Semantic Web – ISWC 2018: 17^{th} International Semantic Web Conference, Monterey, CA, USA, October 8–12, 2018, Proceedings, Part II. Springer, Cham (2018). https://doi.org/10.1007/978-3-030-00668-6_4

[122] M. Ghijsen, J. van der Ham, P. Grosso, C. Dumitru, H. Zhu, Z. Zhao, C. de Laat: A Semantic-Web Approach for Modeling Computing Infrastructures. Computers & Electrical Engineering (2013) https://doi.org/10.1016/j.compeleceng.2013.08.011

[123] Mathieu Lirzin, Béatrice Markhoff: Vers Une Ontologie Des Interactions HTTP. In: 31^{emes} Journées Francophones d'Ingénierie Des Connaissances, Angers, France (2020)

[124] Megan Katsumi, Mark Fox: Defining Activity Specifications in OWL. In: Proceedings of the 8^{th} Workshop on Ontology Design and Patterns (WOP 2017) Co-Located with the 16^{th} International Semantic Web Conference (ISWC 2017), Vienna, Austria (2017)

[125] Marco Rospocher, Chiara Ghidini, Luciano Serafini: An Ontology for the Business Process Modelling Notation. In: Formal Ontology in Information Systems – Proceedings of the Eighth International Conference. IOS Press, Amsterdam, The Netherlands (2014). https://doi.org/10.3233/978-1-61499-438-1-133

[126] Bram Steenwinckel, Pieter Heyvaert, Dieter De Paepe, Olivier Janssens, Sander Vanden Hautte, Anastasia Dimou, Filip De Turck, Sofie Van Hoecke, Femke

Ongenae: Towards Adaptive Anomaly Detection and Root Cause Analysis by Automated Extraction of Knowledge from Risk Analyses. In: 9^{th} International Semantic Sensor Networks Workshop (SSN) (2018)

[127] Phil Blackwood: Gist Jumpstart. https://www.semanticarts.com/gist-jumpstart/ (2023)

[128] Thanos G. Stavropoulos, Dimitris Vrakas, Danai Vlachava, Nick Bassiliades: BOnSAI: A Smart Building Ontology for Ambient Intelligence. In: Proceedings of the 2^{nd} International Conference on Web Intelligence, Mining and Semantics. Association for Computing Machinery, New York, NY, USA (2012). https://doi.org/10.1145/2254129.2254166

[129] Dario Bonino, Fulvio Corno: DogOnt - Ontology Modeling for Intelligent Domotic Environments. In: The Semantic Web - ISWC 2008. Springer, Cham (2008). https://doi.org/10.1007/978-3-540-88564-1_51

[130] Maria Bermudez-Edo, Tarek Elsaleh, Payam Barnaghi, Kerry Taylor: IoT-Lite: A Lightweight Semantic Model for the Internet of Things and Its Use with Dynamic Semantics. Personal and Ubiquitous Computing (2017) https://doi.org/10.1007/s00779-017-1010-8

[131] Irlan Grangel-Gonzalez, Lavdim Halilaj, Gokhan Coskun, Soren Auer, Diego Collarana, Michael Hoffmeister: Towards a Semantic Administrative Shell for Industry 4.0 Components. In: 2016 IEEE Tenth International Conference on Semantic Computing (ICSC). IEEE, USA (2016). https://doi.org/10.1109/ICSC.2016.58

[132] Maxime Lefrançois: Smart Applications REFerence Ontology (SAREF). ETSI (2020)

[133] Krzysztof Janowicz, Armin Haller, Simon Cox, Danh Phuoc, Maxime Lefrançois: SOSA: A Lightweight Ontology for Sensors, Observations, Samples, and Actuators. SSRN Electronic Journal (2018) https://doi.org/10.1016/j.websem.2018.06.003

[134] Mads Holten Rasmussen, Maxime Lefrançois, Georg Ferdinand Schneider, Pieter Pauwels: BOT: The Building Topology Ontology of the W3C Linked Building Data Group. Semantic Web Journal (2020) https://doi.org/10.3233/SW-200385

[135] María Poveda-Villalón, Mari Carmen Suárez-Figueroa, Raúl García-Castro, Asunción Gómez-Pérez: A Context Ontology for Mobile Environments. In: Proceedings of the Second Workshop on Context, Information and Ontologies (2010)

[136] Maxime Lefrançois: Planned ETSI SAREF Extensions Based on the W3C&OGC SOSA/SSN-compatible SEAS Ontology Patterns. In: Workshop on Semantic

Interoperability and Standardization in the IoT (SIS-IoT) (2017)

[137] Noam Ben-Asher, A. Oltramari, R. Erbacher, Cleotilde González: Ontology-Based Adaptive Systems of Cyber Defense. In: 10^{th} Conference on Semantic Technology for Intelligence, Defense, and Security (STIDS) (2015)

[138] Ian Horrocks, Peter F. Patel-Schneider, Harold Boley, Said Tabet, Benjamin Grosof, Mike Dean: SWRL: A Semantic Web Rule Language Combining OWL and RuleML. W3C Member Submission, W3C (2004)

[139] Holger Knublauch, James A. Hendler, Kingsley Idehen: SPIN – Overview and Motivation. W3C Member Submission, W3C (2011)

[140] Adrian Bondy, U. S. R. Murty: Graph Theory. Graduate Texts in Mathematics. Springer, Cham (2008)

[141] Michael Schlichtkrull, Thomas N. Kipf, Peter Bloem, Rianne van den Berg, Ivan Titov, Max Welling: Modeling Relational Data with Graph Convolutional Networks. In: The Semantic Web. Springer, Cham (2018)

[142] Sigaud, O., Buffet, O.: Processus Décisionnels de Markov en Intelligence Artificielle. IC2 - informatique et systèmes d'information, vol. 1 - principes généraux et applications. Lavoisier - Hermes Science Publications, ??? (2008). https://inria.hal.science/inria-00326864

[143] Ferdinand Wagner: Modeling Software with Finite State Machines: A Practical Approach. Auerbach Publications, USA (2006). https://doi.org/10.1201/9781420013641

[144] Gerald H, Sitt Min Oo, Gertjan De Mulder, Michiel Derveeuw, Pieter Heyvaert, Wouter Maroy, Vincent Emonet, kmhaeren, Ben De Meester, Dylan Van Assche, Thomas, ajuvercr: RMLio/RMLStreamer (2022). https://doi.org/10.5281/zenodo.7181800

[145] Anastasia Dimou, Miel Vander Sande, Pieter Colpaert, Ruben Verborgh, Erik Mannens, Rik Van de Walle: RML: A Generic Language for Integrated RDF Mappings of Heterogeneous Data. In: Proceedings of the Workshop on Linked Data on the Web, LDOW 2014, Co-located with the 23^{rd} International World Wide Web Conference (WWW 2014). CEUR-WS.org, Germany (2014)

[146] Pieter Bonte, Riccardo Tommasini, Emanuele Della Valle, Filip De Turck, Femke Ongenae: Streaming MASSIF: Cascading Reasoning for Efficient Processing of IoT Data Streams. Sensors (2018) https://doi.org/10.3390/s18113832

[147] Davide Francesco Barbieri, Daniele Braga, Stefano Ceri, Emanuele Della Valle, Michael Grossniklaus: C-SPARQL: SPARQL for Continuous Querying. In: Proceedings of the 18^{th} International Conference on World Wide Web. Association

for Computing Machinery, New York, NY, USA (2009). https://doi.org/10.1145/1526709.1526856

[148] Guohui Xiao, Davide Lanti, Roman Kontchakov, Sarah Komla-Ebri, Elem Güzel-Kalaycı, Linfang Ding, Julien Corman, Benjamin Cogrel, Diego Calvanese, Elena Botoeva: The Virtual Knowledge Graph System Ontop. In: The Semantic Web – ISWC 2020 (2020). https://doi.org/10.1007/978-3-030-62466-8_17

[149] Pierre-Antoine Champin, Alain Mille, Yannick Prié: Vers Des Traces Numériques Comme Objets Informatiques de Premier Niveau. Intellectica - La revue de l'Association pour la Recherche sur les sciences de la Cognition (ARCo) (2013) https://doi.org/10.3406/intel.2013.1090

[150] Xiangnan Ren, Olivier Curé, Li Ke, Jeremy Lhez, Badre Belabbess, Tendry Randriamalala, Yufan Zheng, Gabriel Kepeklian: Strider: An Adaptive, Inference-Enabled Distributed RDF Stream Processing Engine. Proceedings of the VLDB Endowment (2017) https://doi.org/10.14778/3137765.3137805

[151] S. N. Narayanan, A. Ganesan, K. Joshi, T. Oates, A. Joshi, T. Finin: Early Detection of Cybersecurity Threats Using Collaborative Cognition. In: 2018 IEEE 4^{th} International Conference on Collaboration and Internet Computing (CIC) (2018). https://doi.org/10.1109/CIC.2018.00054

[152] Bram Steenwinckel, Dieter De Paepe, Sander Vanden Hautte, Pieter Heyvaert, Mohamed Bentefrit, Pieter Moens, Anastasia Dimou, Bruno Van Den Bossche, Filip De Turck, Sofie Van Hoecke, Femke Ongenae: FLAGS: A Methodology for Adaptive Anomaly Detection and Root Cause Analysis on Sensor Data Streams by Fusing Expert Knowledge with Machine Learning. Future Generation Computer Systems (2021) https://doi.org/10.1016/j.future.2020.10.015

[153] Ilaria Maresi: A Data Engineer's Guide to Semantic Modelling. Technical report, The Hyve (2020)

[154] ODP: Ontology Design Patterns . org (ODP). http://ontologydesignpatterns.org

[155] Yuan Ren, Artemis Parvizi, Chris Mellish, Jeff Z. Pan, Kees van Deemter, Robert Stevens: Towards Competency Question-Driven Ontology Authoring. In: 11^{th} European Semantic Web Conference (ESWC) (2014). https://doi.org/10.1007/978-3-319-07443-6_50

[156] Bernard Chabot: Le Dicho-Scope et Les Dichotomies Génériques. https://www.linkedin.com/pulse/le-dicho-scope-et-les-dichotomies-g%C3%A9n%C3%A9riques-bernard-chabot/ (2017)

[157] Raphaël Troncy, Antoine Isaac: DOE : une mise en oeuvre d'une méthode

de structuration différentielle pour les ontologies. In: Actes 13^{e} Journées Francophones sur Ingénierie des Connaissances (IC), Rouen (FR) (2002). https://exmo.inria.fr/files/publications/troncy2002a.pdf

[158] Suárez-Figueroa, M.C., Gómez-Pérez, A., Fernández-López, M.: The NeOn Methodology for Ontology Engineering. In: Ontology Engineering in a Networked World. Springer, Cham (2012). https://doi.org/10.1007/978-3-642-24794-1_2

[159] Guarino, N., Welty, C.A.: An Overview of OntoClean. Springer, Cham (2009). https://doi.org/10.1007/978-3-540-92673-3_9

[160] Obitko, M., Snásel, V., Smid, J.: Ontology Design with Formal Concept Analysis. In: International Conference on Concept Lattices and Their Applications (CLA) (2004). https://ceur-ws.org/Vol-110/paper12.pdf

[161] Cimiano, P., Völker, J.: Text2Onto: a framework for ontology learning and data-driven change discovery. In: Proceedings of the 10^{th} International Conference on Natural Language Processing and Information Systems (NLDB). Springer, Cham (2005). https://doi.org/10.1007/11428817_21

[162] Pietrasik, M., Reformat, M.: A Simple Method for Inducing Class Taxonomies in Knowledge Graphs. In: The Semantic Web. Springer, Cham (2020). https://doi.org/10.1007/978-3-030-49461-2_4

[163] Aldo Gangemi, Silvio Peroni: DiTTO: Diagrams Transformation inTo OWL. In: Proceedings of the ISWC 2013 Posters & Demonstrations Track, a Track Within the 12^{th} International Semantic Web Conference (ISWC), Sydney, Australia (2013)

[164] Thematix Partners LLC: Visual Ontology Modeler. Thematix (2013)

[165] Bārzdiņš, J., Bārzdiņš, G., Čerāns, K., Liepiņš, R., Sproģis, A.: UML Style Graphical Notation and Editor for OWL 2. In: Perspectives in Business Informatics Research. Springer, Cham (2010). https://doi.org/10.1007/978-3-642-16101-8_9

[166] Amélie Cordier, Marie Lefevre, Pierre-Antoine Champin, Olivier Georgeon, Alain Mille: Trace-Based Reasoning - Modeling Interaction Traces for Reasoning on Experiences. In: The 26^{th} International FLAIRS Conference (2013)

[167] Amina Annane, Nathalie Aussenac-Gilles, Mouna Kamel: BBO: BPMN 2.0 Based Ontology for Business Process Representation. In: 20^{th} European Conference on Knowledge Management (ECKM). Academic Conferences and publishing limited, Lisbon, Portugal (2019)

[168] Dragan Gašević, Vladan Devedžić: Petri Net Ontology. Knowledge-Based Systems (2006) https://doi.org/10.1016/j.knosys.2005.12.003

[169] Juan C. Vidal, Manuel Lama, Alberto Bugarın: A High-level Petri Net Ontology Compatible with PNML. Petri Net Newsletter (2006)

[170] Johannes Koch, Carlos A. Velasco, Philip Ackermann: HTTP Vocabulary in RDF 1.0. W3c working group note, W3C (2017). https://www.w3.org/TR/HTTP-in-RDF10

[171] Megan Katsumi, Mark Fox: iCity Transportation Planning Suite of Ontologies. Technical report, University of Toronto (2020)

[172] Lionel Tailhardat, Raphaël Troncy, Yoan Chabot: Leveraging Knowledge Graphs For Classifying Incident Situations in ICT Systems. In: 18^{th} International Conference on Availability, Reliability and Security (ARES) (2023). https://doi.org/10.1145/3600160.3604991

[173] Evaggelia Pitoura: Historical Graphs: Models, Storage, Processing. In: Business Intelligence and Big Data. Lecture Notes in Business Information Processing. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-96655-7_4

[174] Russa Biswas, Lucie-Aimée Kaffee, Michael Cochez, Stefania Dumbrava, Theis E. Jendal, Matteo Lissandrini, Vanessa Lopez, Eneldo Loza Mencía, Heiko Paulheim, Harald Sack, Edlira Kalemi Vakaj, Gerard de Melo: Knowledge Graph Embeddings: Open Challenges and Opportunities. DROPS-IDN/v2/document/10.4230/TGDK.1.1.4 (2023) https://doi.org/10.4230/TGDK.1.1.4

[175] Jan Portisch, Heiko Paulheim: Walk This Way! Entity Walks and Property Walks for RDF2vec (2022). https://doi.org/10.48550/arXiv.2204.02777

[176] Federico Bianchi, Pascal Hitzler: On the Capabilities of Logic Tensor Networks for Deductive Reasoning. In: AAAI 2019 Spring Symposium on Combining Machine Learning with Knowledge Engineering (AAAI-MAKE), Stanford University, Palo Alto, California, USA (2019). https://ceur-ws.org/Vol-2350/paper22.pdf

[177] Besold, T.R., Garcez, A., Bader, S., Bowman, H., Domingos, P., Hitzler, P., Kuehnberger, K.-U., Lamb, L.C., Lowd, D., Lima, P.M.V., de Penning, L., Pinkas, G., Poon, H., Zaverucha, G.: Neural-Symbolic Learning and Reasoning: A Survey and Interpretation (2017)

[178] Ryan Riegel, Alexander Gray, Francois Luus, Naweed Khan, Ndivhuwo Makondo, Ismail Yunus Akhalwaya, Haifeng Qian, Ronald Fagin, Francisco Barahona, Udit Sharma, Shajith Ikbal, Hima Karanam, Sumit Neelam, Ankita Likhyani, Santosh Srivastava: Logical Neural Networks (2020). https://doi.org/

10.48550/arXiv.2006.13155

[179] Bronstein, M.M., Bruna, J., Cohen, T., Veličković, P.: Geometric Deep Learning: Grids, Groups, Graphs, Geodesics, and Gauges (2021). https://doi.org/10.48550/arXiv.2104.13478

[180] Dagnely, P., Ruette, T., Tourwé, T., Tsiporkova, E.: Ontology-driven multilevel sequential pattern mining: mining for gold in event logs of photovoltaic plants. In: 2018 International Conference on Intelligent Systems (IS) (2018)

[181] Amal Guittoum, Francois Aïssaoui, Sébastien Bolle, Fabienne Boyer, Noel De Palma: Inferring Threatening IoT Dependencies Using Semantic Digital Twins Toward Collaborative IoT Device Management. In: Proceedings of the 38^{th}ACM/SIGAPP Symposium on Applied Computing. Association for Computing Machinery, New York, NY, USA (2023). https://doi.org/10.1145/3555776.3578573

[182] Jinlin Yang, David Evans, Deepali Bhardwaj, Thirumalesh Bhat, Manuvir Das: Perracotta: Mining Temporal API Rules from Imperfect Traces. In: Proceedings of the 28^{th} International Conference on Software Engineering. Association for Computing Machinery, New York, NY, USA (2006). https://doi.org/10.1145/1134285.1134325

[183] Marc Bouissou: Gestion de la Complexité dans les Études Quantitatives de Sûreté de Fonctionnement de Systèmes. Éditions Tec & Doc, Paris (2008)

[184] William Eberle, Lawrence Holder: Anomaly Detection in Data Represented as Graphs. Intelligent Data Analysis (2007) https://doi.org/10.3233/IDA-2007-11606

[185] Zach Kurtz, Samuel J. Perl: Measuring Similarity between Cyber Security Incident Reports. In: Forum of Incident Response and Security Teams (FIRST Conference) (2017)

[186] Siddharth Bhatia, Bryan Hooi, Minji Yoon, Kijung Shin, Christos Faloutsos: MIDAS: Microcluster-Based Detector of Anomalies in Edge Streams. In: Proceedings of the AAAI Conference on Artificial Intelligence. AAAI Press, New York, NY, USA (2020). https://doi.org/10.1609/aaai.v34i04.5724

[187] Mathieu Garchery, Michael Granitzer: ADSAGE: Anomaly Detection in Sequences of Attributed Graph Edges Applied to Insider Threat Detection at Fine-Grained Level (2020). https://doi.org/10.48550/arXiv.2007.06985

[188] Liz Maida: A Picture Is Worth 1,000 Rows: Visualizing Security Data with Graph Algorithms. https://www.slideshare.net/neo4j/a-picture-is-worth-1000-rows, Online (2021)

[189] Dmitry Vengertsev, Hemal Thakkar: Anomaly Detection in Graph: Unsupervised Learning, Graph-based Features and Deep Architecture. Technical Report, Department of Computer Science, Stanford University (2015)

[190] Serge Romaric Mouafo Tembo, Sandrine Vaton, Jean-Luc Courant, Stephane Gosselin, Michel Beuvelot: Model-Based Probabilistic Reasoning for Self-Diagnosis of Telecommunication Networks: Application to a GPON-FTTH Access Network. Journal of Network and Systems Management (2017) https://doi.org/10.1007/s10922-016-9401-0

[191] Chia-Cheng Yen, Wenting Sun, Hakimeh Purmehdi, Won Park, Kunal Rajan Deshmukh, Nishank Thakrar, Omar Nassef, Adam Jacobs: Graph Neural Network Based Root Cause Analysis Using Multivariate Time-Series KPIs for Wireless Networks. In: NOMS 2022-2022 IEEE/IFIP Network Operations and Management Symposium. IEEE, Budapest, Hungary (2022). https://doi.org/10.1109/NOMS54207.2022.9789858

[192] Sasan Saqaeeyan, Hamid Haj Seyyed Javadi, Hossein Amirkhani: Anomaly Detection in Smart Homes Using Bayesian Networks. KSII Transactions on Internet and Information Systems (2020)

[193] Sherif Saad, Issa Traore: Ontology-Based Intelligent Network Forensics Investigation. In: 19^{th} International Conference on Software Engineering and Data Engineering (SEDE 2010), San Francisco, California, USA (2010)

[194] Mohammed Alzaabi, Andy Jones, Thomas A. Martin: An Ontology-Based Forensic Analysis Tool. In: Annual ADFSL Conference on Digital Forensics, Security and Law, Richmond, Virginia, USA (2013)

[195] Sánchez-Zas, C., Villagrá, V.A., Vega-Barbas, M., Larriva-Novo, X., Moreno, J.I., Berrocal, J.: Ontology-based approach to real-time risk management and cyber-situational awareness. Future Generation Computer Systems (2023) https://doi.org/10.1016/j.future.2022.12.006

[196] Nyoman Juniarta, Miguel Couceiro, Amedeo Napoli, Chedy Raıssi: Sequential Pattern Mining Using FCA and Pattern Structures for Analyzing Visitor Trajectories in a Museum. In: Proceedings of the Fourteenth International Conference on Concept Lattices and Their Applications, Olomouc, Czech Republic (2018)

[197] Nong Ye: A Markov Chain Model of Temporal Behavior for Anomaly Detection. In: Proceedings of the 2000 IEEE Workshop on Information Assurance and Security, United States Military Academy, West Point, NY, USA (2000)

[198] Hari Koduvely: Anomaly Detection through Reinforcement Learning (2018). https://doi.org/10.13140/RG.2.2.33673.29283

[199] Carol Stanton, Gilad Katz, Dawn Song: Isolation Forest for Anomaly

Detection. https://e3s-center.berkeley.edu/wp-content/uploads/2017/08/RET_CStanton-2015.pdf, Berkeley, CA, USA (2015)

[200] Fei Tony Liu, Kai Ming Ting, Zhi-Hua Zhou: Isolation Forest. In: 2008 Eighth IEEE International Conference on Data Mining (2008). https://doi.org/10.1109/ICDM.2008.17

[201] Pankaj Malhotra, Anusha Ramakrishnan, Gaurangi Anand, Lovekesh Vig, Puneet Agarwal, Gautam Shroff: LSTM-based Encoder-Decoder for Multi-sensor Anomaly Detection (2016)

[202] TensorFlow: Introduction aux auto-encodeurs. https://www.tensorflow.org/tutorials/generative/autoencoder (2020)

[203] Pavithra Vijay: Timeseries Anomaly Detection Using an Autoencoder. https://keras.io/examples/timeseries/timeseries_anomaly_detection/ (2020)

[204] Philipp Grashorn, Jonas Hansen, Marcel Rummens: How Airbus Detects Anomalies in ISS Telemetry Data Using TFX. https://blog.tensorflow.org/2020/04/how-airbus-detects-anomalies-iss-telemetry-data-tfx.html (2020)

[205] Maurras Ulbricht Togbe, Mariam Barry, Aliou Boly, Yousra Chabchoub, Raja Chiky, Jacob Montiel, Vinh-Thuy Tran: Anomaly Detection for Data Streams Based on Isolation Forest Using Scikit-Multiflow. In: Computational Science and Its Applications – ICCSA 2020. Springer, Cagliari, Italy (2020). https://doi.org/10.1007/978-3-030-58811-3_2

[206] Myriam Lopez, Marie Beurton-Aimar, Gayo Diallo, Sofian Maabout: Approche de Traitement Des Logs Pour La Prédiction d'erreurs Critiques. Revue des Nouvelles Technologies de l'Information **Extraction et Gestion des Connaissances RNTI-E-37** (2021)

[207] Changgen Li, Liang Guo, Hongli Gao, Yi Li: Similarity-Measured Isolation Forest: Anomaly Detection Method for Machine Monitoring Data. IEEE Transactions on Instrumentation and Measurement (2021) https://doi.org/10.1109/TIM.2021.3062684

[208] Tatsuaki Kimura, Kei Takeshita, Tsuyoshi Toyono, Masahiro Yokota, Ken Nishimatsu, Tatsuya Mori: Network Failure Detection and Diagnosis by Analyzing Syslog and SNS Data: Applying Big Data Analysis to Network Operations. NTT Technical Review (2013)

[209] Halil Ertan: Single Model Based Anomaly Detection for Multi-Item Datasets. https://towardsdatascience.com/single-model-based-anomaly-detection-for-multi-item-datasets-e4dded5eeeb1 (2020)

[210] Jiabao Zhang, Shenghua Liu, Wenting Hou, Siddharth Bhatia, Huawei Shen, Wenjian Yu, Xueqi Cheng: AugSplicing: Synchronized Behavior Detection in Streaming Tensors. In: Proceedings of the AAAI Conference on Artificial Intelligence, Virtual Conference (2021). https://doi.org/10.1609/aaai.v35i5.16595

[211] A. Adriansyah, B. F. van Dongen, W.M.P. van der Aalst: Cost-Based Conformance Checking using the A∗ Algorithm. Technical Report BPM-11-11, BPMcenter.org (2011)

[212] Oleg Mikhail Sheyner: Scenario Graphs and Attack Graphs. PhD thesis, Carnegie Mellon University, Pittsburgh, PA, USA (2004)

[213] Tailhardat, L., Stach, B., Chabot, Y., Troncy, R.: Graphameleon: Relational Learning and Anomaly Detection on Web Navigation Traces Captured as Knowledge Graphs. In: The Web Conf, May 13-17, 2024, Singapore (2024)

[214] Tailhardat, L., Stach, B., Chabot, Y., Troncy, R.: Graphaméléon : apprentissage des relations et détection d'anomalies sur les traces de navigation Web capturées sous forme de graphes de connaissances. In: Plate-Forme Intelligence Artificielle (PFIA), IC Track, July 01-05, 2024, La Rochelle, France (2024)

[215] Vern Paxson: Bro: A System for Detecting Network Intruders in Real-Time. Computer Networks (1999) https://doi.org/10.1016/S1389-1286(99)00112-7

[216] Pedro Alipio, Paulo Carvalho, José Neves: Using CLIPS to Detect Network Intrusions. In: Progress in Artificial Intelligence. Springer, ??? (2003). https://doi.org/10.1007/978-3-540-24580-3_40

[217] Shahaboddin Shamshirband, Nor Anuar, Babak Daghighi, Miss Laiha Mat Kiah, Ahmed Patel, Ajith Abraham: Co-FQL: Anomaly Detection Using Cooperative Fuzzy Q-learning in Network. Journal of Intelligent and Fuzzy Systems (2014) https://doi.org/10.3233/IFS-141419

[218] Pedro Alípio, José Neves, Paulo Carvalho: Automatic Detection of SLS Violation Using Knowledge Based Systems. In: Knowledge-Based Intelligent Information and Engineering Systems. Springer, ??? (2006). https://doi.org/10.1007/11892960_130

[219] Guillaume Brogi, Valerie Viet Triem Tong: TerminAPTor: Highlighting Advanced Persistent Threats through Information Flow Tracking. In: 8^{th} IFIP International Conference on New Technologies, Mobility and Security (NTMS) (2016). https://doi.org/10.1109/NTMS.2016.7792480

[220] Sihem Cherrared, Sofiane Imadali, Eric Fabre, Gregor Gössler: SAKURA a Model Based Root Cause Analysis Framework for vIMS (Poster). In: Proceedings of the 17^{th} Annual International Conference on Mobile Systems, Applications, and Services. ACM, Seoul Republic of Korea (2019). https://doi.org/10.1145/

3307334.3328642

[221] WeBank FinTech: AIOps Series V : RCA Based on Knowledge Graph (2020)

[222] Sihem Cherrared: Fault Management of Programmable Multi-Tenant Networks. PhD thesis, Université Rennes 1 (2020)

[223] Diane Maillot-Tchofo, Ahmed Triki, Maxime Laye, John Puentes: Clustering of Live Network Alarms Using Unsupervised Statistical Models. In: 49^{th} European Conference on Optical Communications (ECOC), Glasgow, Scotland (2023)

[224] Renier van Heerden, Louise Leenen, Barry Irwin: Automated Classification of Computer Network Attacks. In: 2013 International Conference on Adaptive Science and Technology, Pretoria, South Africa (2013). https://doi.org/10.1109/ICASTech.2013.6707510

[225] Renier Pelser van Heerden: A Formalised Ontology for Network Attack Classification. PhD thesis, Rhodes University, Grahamstown, South Africa (2014)

[226] Alfredo Cuzzocrea, Giuseppe Pirrò: A Semantic-Web-Technology-Based Framework for Supporting Knowledge-Driven Digital Forensics. In: Proceedings of the 8^{th} International Conference on Management of Digital EcoSystems. Association for Computing Machinery, Biarritz, France (2016). https://doi.org/10.1145/3012071.3012099

[227] Jorge Martinez-Gil, Georg Buchgeher, David Gabauer, Bernhard Freudenthaler, Dominik Filipiak, Anna Fensel: Root Cause Analysis in the Industrial Domain Using Knowledge Graphs: A Case Study on Power Transformers. Procedia Computer Science (2022) https://doi.org/10.1016/j.procs.2022.01.292

[228] Zhuo Chen, Wen Zhang, Yufeng Huang, Mingyang Chen, Yuxia Geng, Hongtao Yu, Zhen Bi, Yichi Zhang, Zhen Yao, Wenting Song, Xinliang Wu, Yi Yang, Mingyi Chen, Zhaoyang Lian, Yingying Li, Lei Cheng, Huajun Chen: Tele-Knowledge Pre-training for Fault Analysis. In: 39^{th} International Conference on Data Engineering (ICDE) (2023). https://doi.org/10.1109/ICDE55515.2023.00265

[229] Tony Ohmann, Michael Herzberg, Sebastian Fiss, Armand Halbert, Marc Palyart, Ivan Beschastnikh, Yuriy Brun: Behavioral Resource-Aware Model Inference. In: Proceedings of the 29^{th} ACM/IEEE International Conference on Automated Software Engineering - ASE '14. ACM Press, Vasteras, Sweden (2014). https://doi.org/10.1145/2642937.2642988

Appendix A State of the Art – Detailed Materials

This section includes Tables A1 to A4, which provide comprehensive details on the shortlist of 57 references from the literature review on anomaly detection techniques in Section 7, including summaries of each approach. These details support the analyses conducted in that section. Note that some references discuss multiple approaches, which results in the total number of identified approaches exceeding the number of references.

Table A1: Anomaly detection models comparison table – part 1.

Theme	Ref.	Year	Name or short title	Data structures within method
	Short description			
Design		**Graph-based**		
Net-IT	[21]	2019	Measuring Network Reliability ...	Mixed (seq. + graph)
	Probabilistic measure based on network performance through iterative link removal.			
Design		**Knowledge-based**		
Energy systems	[180]	2018	Ontology-driven multilevel sequential ...	Mixed (seq. + graph)
	Mining sequential patterns on a knowledge graph.			
Net-IT	[181]	2023	Inferring Threatening IoT Dependencies ...	Mixed (seq. + tab.)
	Knowledge graph construction for dependency calculus between devices.			
Design		**Model checking**		
Net-IT	[182]	2006	Perracotta	Sequential data
	Association rule learning from software process logs, followed by checking for non-conforming event sequences using activity patterns expressed in rules in a form close to Computation Tree Logic (CTL).			
Design		**Rule-based**		
Energy systems	[183]	2008	Gestion de la complexité dans ...	Graph
	Modeling systems logically and exploring state space to identify potential failure scenarios.			
Detect. & Class.		**Graph-based**		
CyberSec	[184]	2007	GBAD-MDL/P/MPS	Graph
	Detecting node/edge modifications, insertions, and deletions.			
CyberSec	[185]	2017	Measuring Similarity between Cyber ...	Tabular
	Computing similarity between incident reports using graph-based clustering and assignment algorithm.			
CyberSec	[186]	2020	MIDAS	Graph streams
	Detecting sudden bursts of activity with repeated nodes or edges by computing the evolution of event count.			
CyberSec	[187]	2020	ADSAGE	Sequential data
	Classifying the edge representing the predicted next event from audit events as attributed graph edges using a Feedforward Neural Network (FFNN) model for classification and a Recurrent Neural Network (RNN) model for predicting the next event.			
CyberSec	[188]	2021	A Picture is Worth 1,000 Rows	Tabular
	Graph-based clustering to identify related alerts, using centrality measures to prioritize clusters.			
Fraud analytics	[36]	2023	PANG	Graph
	Anomaly detection and fraud analytics in relational or graph-structured data, involving learning graph patterns, ranking and filtering them, graph embedding, and training a classifier using the bag-of-subgraphs concept.			
Net-IT	[189]	2015	DNODA/CNA	Graph
	Analyzing node attribute distribution in relation to global, local, and community perspectives using the Direct Neighbour Outlier Detection Algorithm (DNODA) and Community Neighbor Algorithm (CNA) techniques.			
Net-IT	[190]	2017	Model-Based Probabilistic Reasoning ...	Mixed (sequential data + graph)
	Event classification using a probabilistic model (Bayesian Network) representing network topology and observables, accounting for alarm diffusion phenomena, provided by an expert or learned from incomplete data.			
Net-IT	[191]	2022	Graph Neural Network Based Root ...	Time series
	RCA by constructing a device dependency graph structure using a neural network from performance time series, and performing node classification over the dependency graph using a Graph Convolutional Network (GCN).			
SC-SH	[192]	2020	Anomaly Detection in Smart Homes ...	Sequential data
	Behavior classification on event logs using a probabilistic model (Bayesian Network).			

As per a subset of the evaluation criteria defined in §7.1. In the *Name or short title* column, we indicate the name of the method when provided by the authors, it is the abbreviated title of the paper otherwise. Abbreviations: *Ref.* = bibliographical reference, *SC-SH* = Smart-Cities & Smart-Homes, *n.a.* = non applicable.

Table A2: Anomaly detection models comparison table – part 2.

Theme	Ref.	Year	Name or short title	Data structures within method
	Short description			
Detect. & Class.	**Knowledge-based**			
CyberSec	[37]	2004	Rich Event Representation for ...	Graph
	Classification of computer forensic scenarios by extracting facts from digital event logs and applying expert-made correlation rules using the RETE algorithm.			
CyberSec	[193]	2010	An Ontology-Based Forensic ...	Graph
	Classification and risk analysis of attacks using subsumption-based and graph-traversal-based methods.			
CyberSec	[194]	2013	An Ontology-Based Forensic ...	Graph
	SPARQL-based situation understanding by exploring evidence data from a knowledge graph.			
CyberSec	[137]	2015	PACO	Sequential data (network)
	Binary classification of network traffic using an Instance-Based Learning agent and the PACO OWL-DL ontology.			
CyberSec	[195]	2023	Ontology-based approach to ...	Sequential data
	Binary classification of network sensor data using a Machine Learning model, followed by risk categorization for each alert with a Cyber Threat Intelligence ontology using SWRL rules.			
Generic	[146]	2020	StreamingMASSIF	Sequential data
	Processing of data streams using Semantic Web techniques and Complex Event Processing (CEP).			
Net-IT	[172]	2023	Leveraging Knowledge Graphs ...	Mixed (seq. + tab.)
	Detection of incident situations in network operations data as a knowledge graph using queries that represent detection cases.			
SC-SH	[196]	2018	MRGS/MFCSoFCA+symACS	Sequential data
	Trace classification comparing the similarity of traces to behavior patterns derived from Multi-Resolution Grid-based Segmentation (MRGS) and Multi-Frequency Co-occurrence Similarity (MFCS) analysis, in conjunction with Formal Concept Analysis (FCA).			
Detect. & Class.	**Markov model**			
CyberSec	[197]	2000	A Markov Chain Mo...	Sequential data
	Threshold-based event classification using a transition probability matrix. The anomaly detection threshold is determined based on the low probability of a temporal behavior observed in the recent past.			
Detect. & Class.	**ML-based**			
CyberSec	[198]	2018	RL4AD	Sequential data (network)
	Anomaly detection on sequential data using a hybrid reinforcement learning / deep neural network model.			
Energy systems	[199]	2015	Isolation Forest...	Data points
	Benchmarking the Isolation Forest method over 30+ datasets.			
Generic	[200]	2008	iForest	Data points
	Anomaly detection (outliers) by scoring data points based on their distance from the root of a binary search tree.			
Generic	[201]	2016	EncDec-AD	Time series
	Anomaly likelihood based on the reconstruction error of a time series with respect to a learned normal model, which is a single-layer Long Short Term Memory (LSTM).			
Generic	[202]	2020	Introduction aux auto-encodeurs	Time series
	Anomaly detection using the reconstruction error obtained from an EncDec model. The anomaly detection threshold is defined as one standard deviation above the mean reconstruction error.			
Generic	[203]	2020	Timeseries Anomaly Detection...	Time series
	Anomaly detection time series data using the reconstruction error obtained from a convolutional autoencoder. The anomaly detection threshold is defined as the maximum mean absolute error loss value observed during training.			
Industry 4.0	[204]	2020	How Airbus Detects Anomalies ...	Sequential data
	Anomaly detection on sequential data using the reconstruction error obtained from an LSTM autoencoder.			
Industry 4.0	[205]	2020	IForestASD	Sequential data
	Anomaly detection on sequential data using an extended version of the Isolation Forest technique.			

As per a subset of the evaluation criteria defined in §7.1. In the *Name or short title* column, we indicate the name of the method when provided by the authors, it is the abbreviated title of the paper otherwise. Abbreviations: *Ref.* = bibliographical reference, *SC-SH* = Smart-Cities & Smart-Homes, *n.a.* = non applicable.

Table A3: Anomaly detection models comparison table – part 3.

Theme	Ref.	Year	Name or short title	Data structures within method
	Short description			
Detect. & Class.			**ML-based (contd.)**	
Industry 4.0	[206]	2021	Approche de traitement des ...	Sequential data
	Model-based classification (Bayes, decision tree, deep learning) of data over a sliding-window bag of features.			
Industry 4.0	[207]	2021	SM-iForest	Time series
	Anomalous data segments detected using Isolation Forest and a similarity measure.			
Net-IT	[208]	2013	Network Failure Detection ...	Sequential data
	SysLog logs clustered using mined patterns and non-negative matrix factorization.			
Net-IT	[208]	2013	Network Failure Detection ...	Sequential data
	RCA using user messages from social networks, where the messages are first classified using a Support Vector Machine (SVM) model.			
Net-IT	[209]	2020	Single Model Based Anomaly ...	Time series
	Anomalous data segments detected across multiple time series using an Isolation Forest algorithm applied to z-Scores, which represent deviations of each key performance indicator value from its own patterns.			
Net-IT	[210]	2021	FastTrack/AugSplicing	Graph streams
	Detection of synchronous events in multidimensional data streams based on the identification of dense blocks of data (sub-tensors) and iterative partitioning for near-real-time analysis.			
Net-IT	[172]	2023	Leveraging Knowledge Graphs ...	Mixed (seq. + tab.)
	Detection of incident situations in network operations data as a knowledge graph using a graph embedding approach to learn and classify incident contexts, viewed as subgraphs surrounding the graph nodes that represent incident tickets.			
Detect. & Class.			**Model checking**	
Business process	[211]	2011	Cost-Based Conformance Chec...	Sequential data
	Event logs are replayed on a process model to identify skipped activities and inserted activities.			
CyberSec	[212]	2004	Scenario Graphs and Attack ...	Graph
	Definition of rules in the form of graphs, followed by rule execution for activity detection.			
CyberSec	[213, 214]	2024	Graphameleon	Sequential data (network)
	Classification of abnormal user and system behavior using conformance checking on web navigation trace data represented as a knowledge graph.			
Detect. & Class.			**Rule-based**	
CyberSec	[215]	1999	BRO	Sequential data (network)
	Rule-based alerting on live network traffic using a multi-agent software system.			
CyberSec	[216]	2003	Using CLIPS to Detect Net...	Sequential data (network)
	Rule-based alerting on network traffic by combining the SNORT and CLIPS tools and including string pattern matching, certainty factors and time-stamp operators.			
CyberSec	[217]	2014	Co-FQL	Sequential data (network)
	Fuzzy rule-based classification of network traffic packets, where the appropriate combination of rules is learned and decided with help of Q-Learning (reinforcement learning algorithm to learn the value of an action in a particular state).			
CyberSec	[151]	2018	Cognitive CyberSecurity (CCS)	Graph
	Anomaly detection using a SWRL-based rule set on RDF data, with expert knowledge from a complementary knowledge graph (UCO), and additional analytics (statistical, graph) to enrich the knowledge graph.			
Net-IT	[218]	2006	Automatic Detection of SLS ...	Sequential data
	Rule-based situation classification on event logs using the CLIPS tool.			

As per a subset of the evaluation criteria defined in §7.1. In the *Name or short title* column, we indicate the name of the method when provided by the authors, it is the abbreviated title of the paper otherwise. Abbreviations: *Ref.* = bibliographical reference, *SC-SH* = Smart-Cities & Smart-Homes, *n.a.* = non applicable.

Table A4: Anomaly detection models comparison table – part 4.

Theme	Ref.	Year	Name or short title	Data structures within method
	Short description			
Diagnostic Aid			**Graph-based**	
CyberSec	[219]	2016	TerminAPTor	Mixed (seq. + graph)
	Relational structure discovery between attack alerts using information flows and tag propagation approaches.			
Net-IT	[220]	2019	SAKURA	Mixed (seq. + graph)
	RCA by solving a logical dependency graph with observational values using a Satisfiability Modulo Theory (SMT) solver.			
Net-IT	[221]	2020	AIOps Series V : RCA Based ...	Mixed (seq. + graph)
	Unsupervised out-of-trend alerting using LSTM and probability density on transaction volume data stored in a KG.			
Net-IT	[222]	2020	Fault Management of...	Mixed (seq. + graph)
	RCA by constructing a dependency graph based on prior knowledge of network topology, and then calculating constraints (SAT) based on observed events and their testability.			
Net-IT	[223]	2023	Clustering of Liv...	Mixed (seq. + graph)
	Alarm clustering by pre-filtering events based on network topology, followed by clustering based on inter-arrival times.			
Diagnostic Aid			**Knowledge-based**	
CyberSec	[224]	2013	Automated Classification of Computer ...	Mixed (seq. + graph)
	Situation classification is performed using tableau calculus, leveraging an ontology designed for network attacks.			
CyberSec	[225]	2014	A Formalised Ontology for Network ...	Mixed (seq. + graph)
	Situation classification is performed using tableau calculus, leveraging an ontology designed for network attacks.			
CyberSec	[226]	2016	A Semantic-Web-Technology-...	Graph
	Situation understanding with both SWRL-based and SPARQL-based approaches, using an adhoc ontology and an RDF-based representation of facts and events.			
Energy systems	[227]	2022	Root Cause Analysis in the Industrial ...	Graph
	RCA using both SWRL-based and SPARQL-based approaches on data stored in a KG.			
Industry 4.0	[126]	2018	FOLIO	Tabular
	RCA by annotating and processing event streams using rules, leveraging FMEA mapped onto the FOLIO ontology, and utilizing SWRL rules from Fault Tree Analysis (FTA).			
Net-IT	[228]	2023	KTeleBERT	Mixed (seq. + unstr.)
	RCA using a Graph Convolutional Network (GCN) on network events and recorded network documentation elements stored in a knowledge graph.			
Diagnostic Aid			**Model checking**	
Software engineering	[229]	2014	Perfume	Sequential data
	Association rule learning over logs using Timed Propositional Temporal Logic (TPTL).			

As per a subset of the evaluation criteria defined in §7.1. In the *Name or short title* column, we indicate the name of the method when provided by the authors, it is the abbreviated title of the paper otherwise. Abbreviations: *Ref.* = bibliographical reference, *SC-SH* = Smart-Cities & Smart-Homes, *n.a.* = non applicable.