A framework for anomaly detection and classification in Multiple IoT scenarios

A framework for anomaly detection and classification in Multiple IoT scenarios

Francesco Cauteruccio[1], Luca Cinelli[1], Enrico Corradini[2], Giorgio Terracina[1], Domenico Ursino[2*], Luca Virgili[2], Giancarlo Fortino[3], Antonio Liotta[4], and Claudio Savaglio [3]

[1] DEMACS, University of Calabria,
[2] DII, Polytechnic University of Marche,
[3] DIMES, University of Calabria,
[4] Faculty of Computer Science, Free University of Bozen-Bolzano
[*] Contact Author
{cauteruccio, cinelli, terracina}@mat.unical.it; {e.corradini, l.virgili}@pm.univpm.it;
d.ursino@univpm.it; giancarlo.fortino@unical.it; liotta.antonio@gmail.com;
csavaglio@dimes.unical.it

Abstract

The investigation of anomalies is an important element in many scientific research fields. In recent years, this activity has been also extended to social networking and social internetworking, where different networks interact with each other. In these research fields, we have recently witnessed an important evolution because, beside networks of people, networks of things are becoming increasingly common. IoT and Multiple IoT scenarios are thus more and more studied. This paper represents a first attempt to investigate anomalies in a Multiple IoT scenario (MIoT). First, we propose a new methodological framework that can make future investigations in this research field easier, coherent, and uniform. Then, in the context of anomaly detection in an MIoT, we define the so-called "forward problem" and "inverse problem". The definition of these problems allows the investigation of how anomalies depend on inter-node distances, the size of IoT networks, and the degree centrality and closeness centrality of anomalous nodes. The approach proposed herein is applied to a smart city scenario, which is a typical MIoT. Here, data coming from sensors and social networks can boost smart lighting in order to provide citizens with a smart and safe environment.

Keywords: Anomaly Detection; Internet of Things; Multiple IoT; MIoT; Anomaly Investigation; Forward Problem; Inverse Problem

1 Introduction

In the Concise Oxford Dictionary [1], *anomaly* is defined as "*something that deviates from what is standard, normal, or expected*". If regularities allow investigating the general characteristics of a

[1] Concise Oxford Dictionary - `https://en.oxforddictionaries.com`

complex system, anomalies allow the uncover and analysis of unexpected features that might not be otherwise discovered. For this reason, the detection of anomalies has become very important in data analytics, and is widely investigated both in statistics and machine learning [3, 2, 5]. The relevance of anomaly detection is universally acknowledged, since data anomalies are at basis of significant events and patterns. Example application domains include: privacy and cybersecurity [68, 66]; fault detection [37]; ecological disturbances [25]; communication networks [65]; social media life [26, 56, 61, 67]; and gene regulation [41, 40].

In recent years, anomalies have been widely investigated in social networks to detect fraudulent individuals [55, 6], spammers [59, 28], malicious behavior, and so forth. Even more recently, anomaly detection has been analyzed in contexts where more social networks interact with each other [19], thus going from social networking into social internetworking.

Social internetworking is certainly one of the frontiers of social network analysis, since people tend to have multiple social network accounts and can, thus, become "social bridges". Furthermore, all sorts of networked objects are getting increasingly smart and social, giving rise to the so-called Smart Objects (SOs) and revolutionizing both the Internet of Things (IoT) and the Social Internet of Things (SIoT) [11]. Also, several SIoTs and IoTs cooperate with each other through "bridge" objects, thus generating new architectures, referred to in the literature as Multiple IoT (MIoT) [15].

The detection of anomalies in a single-IoT environment has been widely investigated [16, 69, 14, 44, 23], and many results involving privacy, security and fault detection have been found. However, to the best of our knowledge, no investigation on anomalies and their possible detection in an MIoT has been performed so far.

In this paper, we aim at filling this gap by proposing a new methodological framework for anomaly detection and classification in MIoTs. Our framework models anomalies and the corresponding issues in an MIoT by providing a multi-dimensional view, based on three orthogonal taxonomies: *(i)* presence anomalies vs success anomalies; *(ii)* hard anomalies vs soft anomalies; and *(iii)* contact anomalies vs content anomalies. Each combination of the possible values of these dimensions gives rise to a specific type of anomaly to investigate, for instance the *Presence-Hard-Contact* anomalies. Furthermore, anomaly definitions are orthogonal to specific anomaly detection approaches, past or future, which may be applied (and will be combined) in the context of our framework.

Together with the multi-dimensional taxonomy, another main component of our framework is the extension of conventional methodological frameworks to the MIoT case. Our framework has been conceived to address two problems, known as the "forward problem" and the "inverse problem", respectively. In the forward problem, we aim to analyze the effects that multiple anomalies have onto the MIoT. On the other hand, in the inverse problem, which is traditionally more complex, we aim at detecting the source of the anomalies (i.e., the objects that have generated them) based on the effects that these have on the objects or their connections.

In order to show the possible usage of our framework, we present a case study centered around a smart city. Furthermore, in order to evaluate our framework and extract knowledge, we have conducted a series of tests, which we extensively present in this paper. These allowed us to find several important knowledge patterns about anomalies and their effects in an MIoT. Our most important findings may be summarized as follows: *(i)* the effects of the anomalies of a node rapidly decrease as the distance from the node itself increases; *(ii)* anomalies are less evident in an MIoT than in a single IoT; *(iii)* the

number of anomalous nodes increases as the number of IoTs increases, in a roughly linear way; *(iv)* the outdegree of anomalous nodes has a great impact on the spread of the anomaly over the MIoT; *(v)* closeness centrality is even more important than degree centrality in the spread of anomalies; *(vi)* the computation time necessary for the detection of anomalous nodes is polynomial against the number of MIoT nodes; *(vii)* the time necessary for evaluating the effects of anomalies in an MIoT is quadratic against the number of its nodes.

Summarizing, the main contributions of this paper are the following:

- We present three different anomaly taxonomies, orthogonal to each other, obtaining and formalizing eight kinds of different anomalies.
- We present an approach to evaluate the spread and the effects of an anomaly in an MIoT (forward problem) and another one that, starting from the analysis of the effects of one or more anomalies, aims at detecting the anomalous node(s) (inverse problem).
- We present a case study regarding smart cities, which can benefit from our framework, and illustrate several experiments aimed at evaluating the proposed framework and at deriving many knowledge patterns about anomalies in an MIoT.

The rest of this paper is organized as follows. In Section 2, we examine related literature. In Section 3, we illustrate the MIoT paradigm, which is the reference one for our framework. In Section 4, we present our multi-dimensional taxonomy of anomalies in an MIoT context. In Section 5, we introduce the specialization of the forward and the inverse problems for MIoTs. In Section 6, we illustrate our experiments. Finally, in Section 7, we draw important conclusions and outline possible future developments.

2 Related Work

Anomaly detection has been largely investigated in past literature. Here, anomalies have been defined in very different ways, based on the reference domain and data model. A widely accepted definition of anomaly is the one proposed by Hawkins in [35], where an anomaly is defined as "an observation which deviates so much from other observations as to arouse suspicions that it was generated by a different mechanism". A definition of anomaly specific for social networks can be found in [17], where the authors define anomaly as "an observation which appears to ignore interactions and relationships between individuals and their peers". In [24], anomalies are referred to as "patterns in data that do not conform to a well–defined notion of normal behaviour".

Anomaly detection is an issue largely investigated in past literature. The corresponding research studies can be grouped in several ways. One approach distinguishes these studies into: *(i)* surveys and taxonomies, *(ii)* approaches for anomaly detection in generic networks, *(iii)* approaches for anomaly detection in social networks, and *(iv)* other approaches.

If we consider this classification, our approach belongs to class *(iii)*. In this context, we introduce two main novelties, in that: *(i)* we focus on networks of objects instead of networks of people; *(ii)* we focus on multiple network scenarios instead of single networks. In addition, our methodological

framework introduces two further novelties, namely: *(i)* the definition of three new taxonomies specific for anomaly detection in MIoTs; and *(ii)* the investigation of the so called forward and inverse problems in this research context. Moreover, the study we are presenting is orthogonal to other approaches for anomaly detection in network-based data, since we do not aim at proposing a specific approach to address this last issue.

In the following, in order to give a better overview of the literature, we first examine the four classes of research studies on anomalies and, then, present a table comparing our approach to methods introduced in the literature.

Surveys and taxonomies. Recently, several surveys have proposed structured and comprehensive overviews of anomalies to cope with the need of providing usable taxonomies. A first classification of anomalies can be found in [24], which is considered a pioneering paper in this sense. Besides a formal definition of different kinds of anomalies, the authors highlight the challenges related to anomaly detection. In particular, for each class of anomalies introduced, they focus on existing techniques and application domains. Based on their nature, anomalies have been also classified as Point, Contextual and Collective anomalies. Some applications related to these categories are reported in [4, 45, 54, 43].

A significant amount of work has been carried out on anomaly detection in individual IoTs, as captured by a number of survey papers [16, 63, 14]. On the contrary, to the best of our knowledge, no investigation or categorization of possible anomalies in the context of networks and layered networks (mostly related to MIoTs) has been proposed so far. Works presenting relevant aspects are described in the following.

In [7, 4], the authors investigate anomalies in graph-based environments. Specific analyses of this topic can be found in [10] for social networks, in [29, 31, 39, 34] for intrusion detection, in [58] for traffic modelling, and in [41, 40] for gene regulation.

We characterize anomalies as being either static or dynamic, and as being labelled or unlabelled. In [55], the authors survey the state-of-the-art related to the detection of different types of anomalies in social networks. Here, they show that anomalous users' behaviors in social networks are due to a change in their patterns of interaction or in their ways of interacting with the network, which markedly differ from the ones of their peers. The impact of this anomalous behavior can be observed in the resulting structure, allowing anomalies to be characterized as static or dynamic, labelled or unlabelled. For instance, fraudulent individuals may create a network of collaborations to enhance their reputation in a social network. However, when individuals behave in this way, they show an increased level of interaction in the network and tend to form highly interconnected sub-regions therein.

Anomalies in generic networks. In [59], the authors analyze the detection of e-mail spam in a static, unlabelled network context. In particular, they note that spam and other viral materials are typically sent from a single malicious individual to many targets. As a consequence, detecting a specific star-like structure in a network can be a symptom of malicious behavior. Another approach to spam detection is proposed in [28]. In [6], the authors show that both near-stars and near-cliques are indicators of anomalous behaviors in networks. They focus on anomaly detection in weighted graphs. Their approach can be applied to different contexts, such as intrusion detection, spammer detection, anomalies in social networks, and so forth. They also address the problem of anomaly detection in

static, labeled networks. In this context, they consider some ego-networks, each one centered on an individual and, when the sum over a particular label is disproportionately high with respect to the number of edges in the network, they conclude that the corresponding individual has a potentially anomalous behavior. In [36], a universal coding method for unlabeled graphs is introduced and is adopted for anomaly detection in static, unlabeled graphs.

In [27], the authors propose an approach to anomaly detection in dynamic networks. This exploits the analysis of sub-structures, such as maximal cliques, for detecting community-based anomalies, i.e., unexpected variations of communities. In this work, a community coincides with a maximal clique. This approach considers grown, shrunken, merged, split, born and vanished communities, respectively.

In [47], an approach to detect anomalies on dynamic labeled networks in a big data context is presented. Big data is usually equipped with significant amounts of metadata. This approach exploits both raw data and metadata to detect anomalous events. It is based on the probability of an edge to occur between any two nodes. This probability is a function of the linear combination of node attributes.

Anomalies in social networks. In recent years, social networks have been able to attract the interest of many researchers, who have started to study them from many points of view. A recent guide to research methods, applications and software tools related to social network analysis can be found in [20], while a review of social network analysis problems (including anomaly detection) and related applications is presented in [21]. A review of research methods for figurative language analysis in social networks can be found in [1], while the application of social network analysis to extract critical information after a disaster is considered in [42]. Plenty of applications and software tools are also available on this topic. For example, [64] discusses the integration of heterogeneous social networks; [38] analyzes the search of opinion leaders in social networks; while [9] investigates recommendation techniques in this context.

Recently, some authors have started to study scenarios in which several social networks interact with each other to allow their users to achieve certain goals [19]. In past literature, different terms have been used to refer to this context, including multilayer social networks [17], cross platform online social networks [57], multi social networks [46], and Social Internetworking Scenarios [19]. This is a highly investigated field, since the number of users who simultaneously interact with multiple social networks is constantly growing. For instance, in [17], new forms of anomalies emerging in multi-layer social networks are investigated. In [57], the authors propose an approach that exploits an intelligent-sensing model for analyzing behavioral variations in multiple social networks. In it, controlled faulty data, referred to as cognitive tokens, are intentionally introduced in the information flow for attracting anomalous users. The authors show that the same approach could also be applied to a *single* IoT scenario.

The MIoT environment used in this paper represents the extension to smart objects and the IoTs of social internetworking scenarios [15]. Indeed, users joining multiple social networks can be assimilated to objects belonging to different IoTs, although the data type and nature, and the kind of issues to be addressed, are rather different.

Other approaches Several recent approaches on anomaly detection exploit classification through machine learning-based and/or neural network-based engines [52, 8, 18, 63, 50, 30, 49]. Due to the intrinsic nature of these engines, the corresponding approaches do not construct an explicit model of anomalies. This way of proceeding is complementary and dual with respect to the one adopted in our approach which, indeed, aims at modeling anomalies in new MIoT scenarios.

Classification of our approach After having examined the literature about anomalies, we can compare our approach with the most related ones, which have been introduced above. For this purpose, we consider some comparison properties, namely: *(i)* the ability of handling more networks; *(ii)* the usage of a unified scheme; *(iii)* the ability of managing labeled networks; *(iv)* the ability of handling dynamic networks; *(v)* the exploitation of additional metadata; and *(vi)* the usage of structural properties. Based on these features, our approach compares to the and the most related studies, as shown in Table 1.

	Capability of handling more networks	Usage of a unified scheme	Capability of managing labeled networks	Capability of handling dynamic networks	Exploitation of additional metadata	Usage of structural properties
Our approach	✓	✓	✓	✓	✓	✓
[59]	-	✓	-	-	-	✓
[6]	-	-	✓	-	-	✓
[36]	-	-	-	-	-	✓
[27]	-	-	✓	✓	-	✓
[47]	-	-	✓	✓	✓	-
[17]	✓	-	-	-	-	✓
[57]	✓	-	✓	✓	-	-

Table 1: Comparison between our approach and the most related ones

3 The MIoT paradigm

In this section, we provide an overview of the MIoT paradigm, described in detail in [15], since this is the reference case for our study. An MIoT $\mathcal{M}$ consists of a set of m IoTs. Formally speaking:

$$\mathcal{M} = \{\mathcal{I}_1, \mathcal{I}_2, \cdots, \mathcal{I}_m\} \tag{3.1}$$

where $\mathcal{I}_k$, with $k \in [1, m]$, is a single IoT.

Let o_j be an object of $\mathcal{M}$. We assume that if o_j belongs to $\mathcal{I}_k$ it has an instance ι_{j_k}, representing it in $\mathcal{I}_k$. The instance ι_{j_k} consists of a virtual view (or, better, a software interface) representing o_j in $\mathcal{I}_k$. For example, it provides all the other instances of $\mathcal{I}_k$, and the users interacting with this IoT, with all the necessary information about o_j. The information stored in ι_{j_k} is represented according to the format and the conventions adopted in $\mathcal{I}_k$.

An MIoT $\mathcal{M}$ can also be represented by means of a graph-based notation. In particular, a graph $G_k = \langle N_k, A_k \rangle$ may be associated with an IoT, $\mathcal{I}_k$ of $\mathcal{M}$. In this case:

- N_k is the set of nodes of G_k; there is a node n_{j_k} for each instance $\iota_{j_k} \in \mathcal{I}_k$, and vice versa. Since there is a biunivocal correspondence between a node and an instance, in the following we shall use these two terms interchangeably.
- A_k is the set of the arcs of G_k; there is an arc, $a_{jq_k} = (n_{j_k}, n_{q_k})$ if there exists any form of relationship from n_{j_k} to n_{q_k}.

Finally:

$$\mathcal{M} = \langle N, A \rangle \tag{3.2}$$

Here:

$$N = \bigcup_{k=1}^{m} N_k \qquad\qquad A = A_I \cup A_C \tag{3.3}$$

where:

$$A_I = \bigcup_{k=1}^{m} A_k \qquad\qquad A_C = \{(n_{j_k}, n_{j_q}) | n_{j_k} \in N_k, n_{j_q} \in N_q, k \neq q\} \tag{3.4}$$

A_I is the set of the inner arcs (hereafter, *i-arcs*) of $\mathcal{M}$; they link instances of different objects belonging to the same IoT. A_C is the set of cross arcs (hereafter, *c-arcs*) of $\mathcal{M}$; they link instances of the same object belonging to different IoT. A node connected to at least one c-arc is called *c-node*; otherwise, it is called *i-node*.

In $\mathcal{M}$, an object o_j has associated *a set MD_j of metadata*. Our metadata model refers to the one of the IPSO (Internet Protocol for Smart Object) Alliance[2]. Specifically, MD_j consists of three subsets, namely: *(i)* MD_j^D, i.e., the set of *descriptive metadata*; *(ii)* MD_j^T, i.e., the set of *technical metadata*; *(iii)* MD_j^B, i.e., the set of *behavioral metadata*. All details about these metadata can be found in [15].

Given a pair of instances ι_{j_k} of o_j and ι_{q_k} of o_q in $\mathcal{I}_k$, our model saves the set TrS_{jq_k} of the transactions from ι_{j_k} to ι_{q_k}. It is defined as:

$$TrS_{jq_k} = \{Tr_{jq_{k_1}}, Tr_{jq_{k_2}}, \cdots, Tr_{jq_{k_v}}\} \tag{3.5}$$

A transaction $Tr_{jq_{k_z}} \in TrS_{jq_k}$ is represented as follows:

$$Tr_{jq_{k_z}} = \langle st_{jq_{k_z}}, fh_{jq_{k_z}}, ok_{jq_{k_z}}, ct_{jq_{k_z}} \rangle \tag{3.6}$$

Here:

- $st_{jq_{k_z}}$ denotes the starting timestamp of $Tr_{jq_{k_z}}$.
- $fh_{jq_{k_z}}$ indicates the ending timestamp of $Tr_{jq_{k_z}}$.

[2] `https://www.omaspecworks.org/`

- $ok_{jq_{k_z}}$ denotes whether $Tr_{jq_{k_z}}$ was successful or not; it is set to `true` in the affirmative case, to `false` in the negative one, and to `NULL` if it is still in progress.

- $ct_{jq_{k_z}}$ indicates the set of the content topics considered by $Tr_{jq_{k_z}}$. Specifically, it consists of a set of w keywords:

$$ct_{jq_{k_z}} = \{kw^1_{jq_{k_z}}, kw^2_{jq_{k_z}}, \ldots, kw^w_{jq_{k_z}}\} \tag{3.7}$$

An important subset of TrS_{jq_k} is $TrOkS_{jq_k}$, which stores the successful transactions of TrS_{jq_k}. It is defined as:

$$TrOkS_{jq_k} = \{Tr_{jq_{k_z}} | Tr_{jq_{k_z}} \in TrS_{jq_k}, ok_{jq_{k_z}} = \texttt{true}\} \tag{3.8}$$

In other words, this set comprises all the transactions through which ι_{q_k} gave a positive answer to a request of ι_{j_k}, thus providing this last one with services, information or data it required.

Now, we can define the set TrS_{j_k} of the transactions activated by ι_{j_k} in $\mathcal{I}_k$. Specifically, let $\iota_{1_k}, \iota_{2_k}, \cdots, \iota_{w_k}$ be all the instances belonging to $\mathcal{I}_k$. Then:

$$TrS_{j_k} = \bigcup_{q=1..w, q \neq j} TrS_{jq_k} \tag{3.9}$$

This means that the set TrS_{j_k} of the transactions of an instance ι_{j_k} is given by the union of the sets of the transactions from ι_{j_k} to all the other instances of $\mathcal{I}_k$.

We should note that, herein, we have reported only those aspects of the MIoT paradigm that are strictly necessary for this paper. The interested reader can find further details in [15].

We can now introduce the concept of neighborhood of an instance ι_{j_k} in $\mathcal{I}_k$. Specifically, the neighborhood Nbh_{j_k} of ι_{j_k} is defined as:

$$Nbh_{j_k} = ONbh_{j_k} \cup INbh_{j_k} \tag{3.10}$$

where:

$$ONbh_{j_k} = \{n_{q_k} | (n_{j_k}, n_{q_k}) \in A_I, |TrS_{jq_k}| > 0\} \quad INbh_{j_k} = \{n_{q_k} | (n_{q_k}, n_{j_k}) \in A_I, |TrS_{qj_k}| > 0\} \tag{3.11}$$

In other words, Nbh_{j_k} comprises those instances directly connected to ι_{j_k} through an incoming or an outgoing arc, which shared at least one transaction with it.

Finally, we can define the concept of neighborhood of an i-arc $a_{jq_k} = (n_{j_k}, n_{q_k}) \in A_I$. Specifically, the neighborhood Nbh_{jq_k} of the i-arc a_{jq_k} is defined as:

$$Nbh_{jq_k} = ONbh_{jq_k} \cup INbh_{jq_k} \tag{3.12}$$

where:

$$ONbh_{jq_k} = \{(n_{q_k}, n_{r_k}) | (n_{q_k}, n_{r_k}) \in A_I\} \quad INbh_{jq_k} = \{(n_{l_k}, n_{j_k}) | (n_{l_k}, n_{j_k}) \in A_I\} \tag{3.13}$$

Hence, $ONbh_{jq_k}$ contains all the arcs of A_I having n_{q_k} as source node, whereas $INbh_{jq_k}$ comprises all the arcs of A_I having n_{j_k} as target node.

4 Modeling Anomalies in an MIoT

In this section, we propose a model allowing for the representation and management of anomalies in MIoTs. The core of our model consists of some possible taxonomies characterizing anomalies in this scenario. Each one will correspond to different analysis viewpoints. Borrowing a terminology typical in data analysis, these taxonomies can be seen as different dimensions of a multi-dimensional model, through which the fact "anomalies in an MIoT" can be investigated. In this paper, we consider three of these taxonomies, namely: *(i)* presence anomalies vs success anomalies; *(ii)* hard anomalies vs soft anomalies; *(iii)* contact anomalies vs content anomalies. However, we do not exclude that other taxonomies may also be possible in future works.

Continuing with the analogy between our taxonomies and the dimensions of a multi-dimensional model, we have that each combination of the possible values of these dimensions gives rise to a specific type of anomaly to study. Therefore, we have the *Presence-Hard-Contact Anomalies*, the *Success-Hard-Content Anomalies*, and so on. In the following subsections, we briefly illustrate each taxonomy and, then, provide a formalization for some types of combined anomalies. We point out again that the description of our taxonomies is orthogonal to specific anomaly detection techniques. In order to keep the formalization as clear as possible, we will focus on a simple anomaly detection scheme based on frequencies. However, more complex detection schemes may certainly be applied to our taxonomies.

4.1 Definition of anomaly taxonomies

4.1.1 Presence Anomalies vs Success Anomalies

A *presence anomaly* denotes that there is a strong variation (i.e., *increase* or *decrease*) in the number of transactions carried out from an instance ι_{j_k} to an instance ι_{q_k} in a unit of time. A *success anomaly* shows that, although there is no presence anomaly from ι_{j_k} to ι_{q_k}, there is a strong *decrease* in the number of *successful* transactions from ι_{j_k} to ι_{q_k} in a unit of time.

4.1.2 Hard Anomalies vs Soft Anomalies

A *hard anomaly* indicates that the frequency of successful transactions carried out from an instance ι_{j_k} to an instance ι_{q_k} is higher than (or lower than) a certain threshold. A *soft anomaly* happens when the frequency of the (successful) transactions ranges between the maximum and the minimum thresholds but, for several consecutive instances of time, it is higher (resp., lower) than the mean of these two thresholds and it shows a monotone increasing (resp., decreasing) trend. The rationale underlying this taxonomy is that hard anomalies are indicators of faults, whereas soft anomalies are indicators of a slow, but constant, degradation. Soft anomalies are extremely precious in applications such as predictive maintenance.

4.1.3 Contact Anomalies and Content Anomalies

A *contact anomaly* from an instance ι_{j_k} to an instance ι_{q_k} considers only the presence or the absence of transactions. By contrast, a *content anomaly* takes the content exchanged in the corresponding

transactions into account[3]. Here, we assume that we are capable of identifying possible synonymies or homonymies relating terms. This is a well-known problem in the cooperative information system research field and several thesauruses have been proposed for this purpose. In this paper, unless otherwise specified, we will refer to Babelnet [48], which is among the most advanced thesauruses. As far as content anomalies are concerned, a reference content set, consisting of some keywords, is necessary for verifying variations with respect to the content of the involved transactions. Two variants of content anomalies can be considered, namely: *(i)* the *strict* content anomalies, where the whole set of the reference keywords must be present in the involved transactions, and *(ii)* the *loose* content anomalies, where at least one of the reference keywords must be present therein.

4.2 Formalization of anomalies

The combination of the three taxonomies introduced above gives rise to eight possible kinds of anomaly. In the following, we provide the formal definition for representative cases. We recall that, for the sake of clarity, in these definitions we consider frequencies as the basic factor for anomaly detection. However, we point out that, even if frequencies are a well-accepted and widely adopted factor, even more complex factors could easily be incorporated into our taxonomies.

In the next subsections, we present a formalization of a representative selection of the eight anomaly types, providing the method for computing their anomaly degrees. We have not included the formalization for all cases, due to brevity. Yet, their definition would be analogous and straightforward.

The kinds of anomaly that we formalize below include: *(i)* Presence-Hard-Contact anomalies, *(ii)* Success-Hard-Contact anomalies, *(iii)* Presence-Soft-Contact anomalies, and *(iv)* Presence-Hard-Content anomalies. In many of these definitions, the variable "time" plays a key role.

4.2.1 Presence-Hard-Contact Anomalies

Let t be a time instant and let Δt be a time interval (consisting of one or more time units). The frequency $TrFr_{jq_k}(t, \Delta t)$ of the transactions from ι_{j_k} to ι_{q_k} can be defined as follows:

$$TrF_{jq_k}(t, \Delta t) = \frac{|\{Tr_{jq_{k_z}} \mid Tr_{jq_{k_z}} \in TrS_{jq_k}, st_{jq_{k_z}} \geq t, fh_{jq_{k_z}} \leq (t + \Delta t)\}|}{\Delta t} \tag{4.1}$$

In other words, TrF_{jq_k} is given by the ratio between the number of transactions from ι_{j_k} to ι_{q_k} exchanged in the time interval $[t, t + \Delta t]$ to the length of this time interval (i.e., Δt).

We say that there is a Presence-Hard-Contact anomaly from ι_{j_k} to ι_{q_k} in the time interval $[t, t+\Delta t]$ if:

- TrF_{jq_k} is higher than a certain threshold th_{max}, in which case the anomaly degree is defined as $\alpha_{jq_k}(t, \Delta t) = \frac{TrF_{jq_k}(t,\Delta t) - th_{max}}{th_{max}}$, or
- TrF_{jq_k} is lower than a certain threshold th_{min} and this inequality does not hold in the time instants preceding t. This last condition is necessary to avoid that the lack of transactions from ι_{j_k} to ι_{q_k} is erroneously interpreted as a presence anomaly, as it would the case for instance when

[3]Recall that, given a transaction $Tr_{jq_{k_z}}$, the corresponding content $ct_{jq_{k_z}}$ consists of a set of w keywords.

two instances have never performed transactions between them in the past. In this case, the anomaly degree is defined as $\alpha_{jq_k}(t, \Delta t) = \frac{th_{min} - TrF_{jq_k}(t,\Delta t)}{th_{min}}$.

If no Presence-Hard-Contact anomaly is detected, $\alpha_{jq_k}(t, \Delta t)$ is set to 0.

Here and in the following, the thresholds th_{max} and th_{min} can either be static or are dynamically computed over the previous observations. For instance, they could be computed considering both the mean and the standard deviation observed for TrF_{jq_k} in a predefined period of time. However, their actual definition depends on the application domain.

Presence-Hard-Contact anomalies focus on anomalies detected in the number of transactions (*presence*) occurring between two *instances* in an MIoT without considering the content they share (*contact*) and focusing on sharp variations of observed values (*hard*).

Their detection could be particularly relevant, for example, to identify faults concerning the ability of an MIoT object to send data. This may happen, for instance, because an object is no longer working.

Here and in the following, thanks to the concept of MIoT, anomalies between pairs of instances can be used to compute anomalies between the corresponding pairs of objects. In particular, given two objects o_j and o_q, let $\mathcal{IS}_{jq}$ be the set of IoTs containing instances of both o_j and o_q connected by an *i-arc*. The anomaly degree $\alpha_{jq}(t, \Delta t)$ between the pair of objects o_j and o_q in an MIoT can be defined as:

$$\alpha_{jq}(t, \Delta t) = \frac{\sum_{\mathcal{I}_k \in \mathcal{IS}_{jq}} \alpha_{jq_k}(t, \Delta t)}{|\mathcal{IS}_{jq}|} \tag{4.2}$$

This way of computing anomalies between pairs of objects in an MIoT, starting from the anomalies of the corresponding pairs of instances, is valid for all kinds of anomalies.

4.2.2 Success-Hard-Contact Anomalies

Similarly to what we have done for Presence-Hard-Contact anomalies, we first define the frequency $TrOkF_{jq_k}(t, t + \Delta t)$ of the transactions from ι_{j_k} to ι_{q_k} that occurred successfully in the time interval $[t, t + \Delta t]$ as:

$$TrOkF_{jq_k}(t, \Delta t) = \frac{|\{Tr_{jq_{k_z}} \mid Tr_{jq_{k_z}} \in TrOkS_{jq_k}, st_{jq_{k_z}} \geq t, fh_{jq_{k_z}} \leq (t + \Delta t)\}|}{\Delta t} \tag{4.3}$$

Now, we can say that, in the time interval $[t, t + \Delta t]$, there is a Success-Hard-Contact anomaly if:

- there is no Presence-Hard-Contact anomaly in the same time interval;
- $TrOkF_{jq_k}$ is lower than a certain threshold th'_{min}.

In this case, the anomaly degree is defined as $\alpha_{jq_k}(t, \Delta t) = \frac{th'_{min} - TrOkF_{jq_k}(t,\Delta t)}{th'_{min}}$. Otherwise, $\alpha_{jq_k}(t, \Delta t) = 0$.

Success-Hard-Contact anomalies are very similar to Presence-Hard-Contact anomalies. However, they focus on the fraction of successful transactions occurring between two instances in an MIoT (*success*); they disregard the content exchanged by transactions (*contact*) and focus on sharp variations of observed values (*hard*).

The detection of this kind of anomaly might be particularly relevant, for example, in recognizing possible difficulties of an MIoT object to deliver requested data. Differently from the previous case, this may happen because there is an issue in the network rather than in the object itself.

4.2.3 Presence-Soft-Contact Anomalies

Let t be a time instant, let Δt be a time interval and let τ be a positive integer representing the number of time units after t into consideration (generally, $\tau \gg \Delta t$), and let $th_{avg} = \frac{th_{min}+th_{max}}{2}$. We can say that, in the time interval $[t, t+\tau]$, there is a Presence-Soft-Contact anomaly if, for each time instant θ such that $t \leq \theta \leq t+\tau$, the following conditions hold:

- $th_{min} \leq TrF_{jq_k}(\theta, \Delta t) \leq th_{max}$, which implies that no Presence-Hard-Contact anomaly exists in the time interval into consideration;
- $TrF_{jq_k}(\theta, \Delta t) > th_{avg}$ (resp., $TrF_{jq_k}(\theta, \Delta t) < th_{avg}$), which denotes that the frequency of the transactions from ι_{j_k} to ι_{q_k} is always higher (resp., smaller) than the average between th_{min} and th_{max};
- $TrF_{jq_k}(\theta+1, \Delta t) \geq TrF_{jq_k}(\theta, \Delta t)$ (resp., $TrF_{jq_k}(\theta+1, \Delta t) \leq TrF_{jq_k}(\theta, \Delta t)$), which implies that the frequency of the transactions from ι_{j_k} to ι_{q_k} is monotonically increasing (resp., decreasing) in the time interval Δt of interest.

If an anomaly is detected, the corresponding anomaly degree $\alpha_{jq_k}(t, \Delta t)$ is set to $\alpha_{jq_k}(t, \Delta t) = \frac{|TrF_{jq_k}(t+\tau, \Delta t) - th_{avg}|}{th_{avg}}$. Otherwise, $\alpha_{jq_k}(t, \Delta t) = 0$.

Presence-Soft-Contact anomalies focus on a smooth (*soft*) decrease in the number of all (*presence*) the transactions exchanged between two instances of an MIoT, without considering the exchanged content (*contact*).

The detection of this kind of anomaly may be useful in identifying a slowly but constantly changing behavior of an object. For instance, it could regard an object that is wearing out, an equipment whose battery has a very low charge level, and so forth.

4.2.4 Presence-Hard-Content Anomalies

Let $\overline{ct}$ be a content consisting of (presumably very few) keywords. We define the set $sTrCtS_{jq_k}(\overline{ct})$ of the transactions from ι_{j_k} to ι_{q_k} *strictly adherent* to $\overline{ct}$, i.e., the set of the transactions from ι_{j_k} to ι_{q_k} that contain *all the keywords* of $\overline{ct}$ as follows:

$$sTrCtS_{jq_k}(\overline{ct}) = \{Tr_{jq_{k_z}} \mid Tr_{jq_{k_z}} \in TrS_{jq_k}, \overline{ct} \subseteq ct_{jq_{k_z}}\} \tag{4.4}$$

As previously pointed out, here we assume that we are capable of identifying possible synonymies or homonymies relating a term of $\overline{ct}$ with a term of $ct_{jq_{k_z}}$. For this purpose, we use Babelnet [48].

Consider, now, a content $\overline{ct}$ consisting of some keywords. We define the set $lTrCtS_{jq_k}(\overline{ct})$ of the transactions from ι_{j_k} to ι_{q_k} that are *loosely adherent* to $\overline{ct}$, i.e., the set of the transactions from ι_{j_k} to ι_{q_k} that contain *at least one keyword* of $\overline{ct}$ as follows:

$$lTrCtS_{jq_k}(\overline{ct}) = \{Tr_{jq_{k_z}} \mid Tr_{jq_{k_z}} \in TrS_{jq_k}, (\overline{ct} \cap ct_{jq_{k_z}}) \neq \emptyset\} \quad (4.5)$$

Let t be a time instant and let Δt be a time interval. By applying the same approach described for Presence-Hard-Contact anomalies, it is possible to define the frequency $sTrCtF_{jq_k}(\overline{ct})$ (resp., $lTrCtF_{jq_k}(\overline{ct})$) of the transactions from ι_{j_k} to ι_{q_k} strictly (resp., loosely) adherent to $\overline{ct}$. Then, it is possible to state that, in the time interval $[t, t + \Delta t]$, there is a strict (resp., loose) Presence-Hard-Content anomaly from ι_{j_k} to ι_{q_k} against $\overline{ct}$ if:

- $sTrCtF_{jq_k}(\overline{ct})$ (resp., $lTrCtF_{jq_k}(\overline{ct})$) is higher than a certain threshold th_{max}, or
- $sTrCtF_{jq_k}(\overline{ct})$ (resp., $lTrCtF_{jq_k}(\overline{ct})$) is lower than a certain threshold th_{min} and this inequality does not hold in the time instants preceding t.

Analogously to what we have done for Presence-Hard-Contact anomalies, if the first condition is verified, the anomaly degree $\alpha_{jq_k}(t, \Delta t)$ can be defined as $\alpha_{jq_k}(t, \Delta t) = \frac{sTrCtF_{jq_k}(\overline{ct}) - th_{max}}{th_{max}}$, for strictly adherent anomalies, and $\alpha_{jq_k}(t, \Delta t) = \frac{lTrCtF_{jq_k}(\overline{ct}) - th_{max}}{th_{max}}$, for loosely adherent ones. Instead, if the second condition is verified, then $\alpha_{jq_k}(t, \Delta t) = \frac{th_{min} - sTrCtF_{jq_k}(\overline{ct})}{th_{min}}$, for strictly adherent anomalies, and $\alpha_{jq_k}(t, \Delta t) = \frac{th_{min} - lTrCtF_{jq_k}(\overline{ct})}{th_{min}}$ for loosely adherent ones. $\alpha_{jq_k}(t, \Delta t) = 0$ in all the other cases.

Presence-Hard-Content anomalies focus on sharp variations (*hard*) in the number of transactions (*presence*) exchanged between two instances in an MIoT, with regard to a certain set of contents (*content*).

The study of content variations paves the way to a wide variety of analyses, ranging from variations in the interests of a user who is adopting the MIoT objects, to variations in the sentiment of a user on a specific topic/service provided through the MIoT objects.

The other kinds of anomaly, whose formalization we have not reported in this paper because they are very similar to the ones considered above, would provide four further viewpoints of the possible anomalies existing in an MIoT. It would be straightforward to see how these extra anomalies would allow us to model other possible real-world cases, which shows the generic applicability of our approach (three taxonomies and a multi-dimensional perspective).

5 Investigating the origins and effects of anomalies in an MIoT

After providing a multi-dimensional taxonomy of the possible anomalies present in an MIoT, in this section we aim at investigating their origins and effects. For this purpose, we address two problems that, according to what happens in several other research fields, we dubbed "forward problem" and "inverse problem", respectively. In the forward problem, given one or more anomalies, we aim at analyzing their effects on an MIoT. In the inverse problem, which is traditionally more complex than the forward one, given the effects of one or more anomalies on the nodes and the arcs of an MIoT, we aim at detecting the origin(s) of them, i.e., the node(s) or the arc(s) from which anomalies have started.

5.1 Forward Problem

As previously pointed out, this problem aims at understanding the effects that one or more anomalies have on the nodes of an MIoT. In the following, we will investigate the forward problem for one kind of anomaly, namely the Presence-Hard-Contact anomaly. However, all our results can be extended to all the other cases introduced in Section 4.

First, given a node n_{j_k} of an IoT $\mathcal{I}_k$, along with the anomaly degrees of its outgoing arcs, in the forward problem we want to compute the overall effects of these anomalies over the corresponding IoT, $\mathcal{I}_k$. Specifically, the degree $\delta_{j_k}(t, \Delta t)$ of the anomalies of n_{j_k} in the time instant t and in the time interval Δt depends on the number of nodes belonging to $ONbh_{j_k}$ and, for each of these nodes n_{q_k}, on the degree $\delta_{q_k}(t, \Delta t)$ of the anomalies involving it and on the anomaly degrees measured for the corresponding arcs.

We wish to observe that, by saying that the degree of the anomalies of a node n_{j_k} recursively depends on the degree of the anomalies of the nodes belonging to $ONbh_{j_k}$, we introduce a way of proceeding that is similar to the one underlying the definition of the PageRank [51]. Thus, to compute δ_{j_k}, it is possible to adapt the formula for the computation of the PageRank to our scenario. Specifically:

$$\delta_{j_k}(t, \Delta t) = \gamma + (1 - \gamma) \cdot \frac{\sum_{n_{q_k} \in ONbh_{j_k}} \delta_{q_k}(t, \Delta t) \cdot \alpha_{jq_k}(t, \Delta t)}{\sum_{n_{q_k} \in ONbh_{j_k}} \alpha_{jq_k}(t, \Delta t)} \tag{5.1}$$

This formula says that the degree $\delta_{j_k}(t, \Delta t)$ of the anomalies of n_{j_k} in the time instant t and in the time interval Δt is obtained by summing two components:

- The former component, γ, is the damping factor generally existing in each approach based on PageRank. It ranges in the real interval [0,1] and denotes the minimum absolute anomaly degree that can be assigned to a node of the MIoT.

- The second component, is a weighted sum of the anomaly degree $\delta_{q_k}(t, \Delta t)$ of the nodes n_{q_k} directly connected to n_{j_k} and, therefore, belonging to $ONbh_{j_k}$. The weight of each anomaly degree $\delta_{q_k}(t, \Delta t)$ is given by the value of the parameter α_{jq_k}, which considers the fraction of anomalous transactions performed from n_{j_k} to n_{q_k}.

In this formula, $\delta_{j_k}(t, \Delta t)$ ranges in the real interval [0,1].

The above formula allows us to determine the effects of a faulty node over the corresponding IoT, and consequently on the whole MIoT (as will become clearer next). However, we observe that the current formalization is valid only in the presence of a single faulty node. When multiple nodes simultaneously exhibit some anomalous behavior in one IoT (of the MIoT), our approach fails to distinguish among the contributions of each anomaly, particularly when the effects are measured in a single node. We wish to point out that this is our very first attempt to investigate MIoT anomalies, proposing a method to evaluate their effects. Our next priority as a follow-up of the present study, will be extending our method accordingly.

Having investigated the effects of an anomaly of an *instance* in an IoT, we can now exploit the features of the MIoT paradigm to analyze the effects of an anomaly of an *object* in an MIoT. In

particular, the anomaly degree $\delta_j(t, \Delta t)$ of an object o_j can be computed starting from the anomaly degrees of its instances. Specifically, given the set $\mathcal{IS}_j$ of the IoT containing instances of o_j, $\delta_j(t, \Delta t)$ can be computed as:

$$\delta_j(t, \Delta t) = \frac{\sum_{\mathcal{I}_{j_k} \in \mathcal{IS}_j} \delta_{j_k}(t, \Delta t)}{|\mathcal{IS}_j|} \tag{5.2}$$

We observe that the value of $\delta_j(t, \Delta t)$, if compared with the one of $\delta_{j_k}(t, \Delta t)$, can provide very useful information. In particular, if $\delta_j(t, \Delta t)$ is very similar to $\delta_{j_k}(t, \Delta t)$ for each IoT $\mathcal{I}_{j_k} \in \mathcal{IS}_j$, we can conclude that o_j is really a source of anomaly. Instead, if the standard deviation of $\delta_j(t, \Delta t)$ is high, then we can conclude that o_j is involved in, or affected by, some anomalies in one or more IoTs, but not in some other ones.

5.2 Inverse Problem

As previously pointed out, the inverse problem is traditionally more complex than the forward one. For this reason, in this paper, we will focus only on the simplest scenario, i.e., the case in which there is only one anomaly in the MIoT. In the future, we plan to extend our investigation to more complex scenarios. Let $a_{jq_k} = (n_{j_k}, n_{q_k})$ be an i-arc of an MIoT presenting an anomaly whose origin is not known. In the inverse problem we want to detect this origin.

First of all, we must verify if the origin of the anomaly is just a_{jq_k}. For this purpose, we consider the "siblings" of a_{jq_k}, i.e., the other arcs having n_{j_k} as the source node and the other arcs having n_{q_k} as the target node. If none of these present anomalies, then it is possible to conclude that a_{jq_k} is the origin of the observed anomaly and that this last one did not affect other nodes or arcs of the MIoT. In this case, the inverse problem has been solved and the investigation terminates.

However, the situation described above is very particular and, also, quite rare. More typically, anomalies tend to affect multiple nodes and arcs. In that case, given an anomaly found in an arc a_{jq_k}, in order to detect its origin, the first step consists in computing the anomaly degrees of n_{j_k} and n_{q_k} and to choose the maximum between the two. This becomes the current node under investigation.

At this point, an iterative process, aiming at finding the origin of the observed anomaly, is activated. During each step of this process, we apply the PageRank-based formula for the computation of the anomaly degree of a node, as discussed in Section 5.1, to all the nodes of the $ONbh$ and the $INbh$ of the current node. After this, we select the node having the maximum anomaly degree. If the degree of this node is higher than the one of the current node, it becomes the new current node and a new iteration starts. Otherwise, our approach concludes that the current node is the origin of the anomaly under consideration.

Clearly, the approach described above is greedy and, therefore, must be intended as a heuristic that could return a local maximum, instead of a global one. However, it is possible to apply to this approach all the techniques for improving the accuracy of a greedy approach already proposed in past literature, spanning from meta heuristics, such as hill climbing [53], to evolutionary optimization algorithms [60].

For instance, if the MIoT is not excessively large, it could be possible to compute the anomaly degree of all its nodes by applying the PageRank-based approach described in Section 5.1. In this case,

the node having the maximum value of anomaly degree would be selected as the anomaly origin. This would correspond to applying an approach returning the optimum solution to the inverse problem, instead of one returning an approximate solution.

On the opposite extreme, if the network is very large, and the anomaly is affecting a vast portion of it, the greedy approach may be prohibitive. In this case, we will need to find an additional way to stop the iterative process, particularly when resources are limited and the process does not stop because, at each iteration, it continues to return a new current node with an anomaly degree higher than the one of the previous iteration. For instance, we could define a maximum number of iterations or a minimum increase of the anomaly degree necessary to activate a further iteration. Furthermore, this required minimum increase could be dynamic and could vary based on the number of steps already performed.

We conclude this section with an important consideration. Since this is our first paper that investigates the inverse problem, we had the necessity to limit our analysis to only one case, i.e., the one in which, in a certain time instant, there is only one anomaly in the MIoT. If at a given time instant, there are more anomalies in the MIoT, the search of the corresponding origins becomes much more complex, because the anomalies could interfere with each other. These interferences could make the search of the anomaly sources extremely complex.

For instance, we argue that, in presence of two anomalies whose source nodes are not known, in case these two nodes were relatively close to each other, the examination of the anomaly degree of their neighbors could be extremely beneficial. In fact, in this scenario, some of these neighbors are influenced only by one anomaly; other ones are influenced only by the other anomaly; a third group of neighbors is influenced by both anomalies; finally, a fourth group is not influenced by any anomalies. By deeply analyzing what happens in these four groups of nodes, it could be possible to derive precious information leading us to identify the sources of the two anomalies. In the future, we plan to conduct specific and accurate investigations about this case, and several other ones possibly characterizing the inverse problem.

6 Use Case and Experiments

6.1 A smart city use case

All of the devices installed in urban infrastructures, such as smart lighting systems and traffic management ones, contribute to the ecosystem of a so called *smart community*. This last one integrates a series of technological solutions for the definition and implementation of innovative models for the smart management of urban areas. One of the main challenges of the next generation of Information and Communication Technologies (ICT) applied to smart communities is the collection, integration and exploitation of information gathered from heterogeneous data sources, including autonomous smart resources, like SO, sensors, surveillance systems, etc., and human resources, such as posts in social networks. Another key challenge is the application of artificial intelligence tools, such as the ones based on automated reasoning, to advance state-of-the-art in smart community management [22].

The use case we focus on in this section refers to a smart lighting system in a smart city. In particular, we consider a data-centric platform integrated in a smart city environment, in which data

coming from sensors and social networks can boost smart lighting, by operating and tuning different smart lighting objects located in the smart city area. The aim of the whole system is to provide citizens with a smart and safe environment.

Data are gathered from three different main sources, namely sensors, social networks and alerts exchanged among citizens on a dedicated social platform. Sensors data are gathered from a set of sensors installed on each smart lamp and handle different measures, such as temperature and humidity, but also several events, such as the presence of a person or the presence of rain. Sensors and smart lamps are organized in a Wireless Sensor Area Network (WSAN). Social networks data include geo-localized tweets from Twitter and posts from specific Facebook pages and are generated by smart personal devices.

All these data are stored in a data lake, which is directly accessed by a data mining module. This last module includes both sentiment analysis and anomaly detection tasks. The former focuses on the analysis of the data gathered from social posts. A polarity score, i.e., a positiveness/negativeness degree, is assigned to each keyword that can be extracted from a post, and is used to intercept crucial information from the citizens moving around the city. In order to unambiguously single out significant information for the application context, keywords are mapped onto a specific urban taxonomy; this task is also carried out with the support of Babelnet [48]. Furthermore, thanks to the geo-localization of posts, information regarding a specific area of the smart city can be analyzed and assigned to the correct area.

Some data mining tasks are also carried out in order to identify, among other things, situations requiring a variation in the intensity of illumination for some area, for instance because of a variation in the security level perceived by citizens therein. Each smart lamp can communicate with neighboring ones in order to report variations in lighting parameters, as received by the mining module.

Anomaly detection works on both temporal data, gathered from sensors, and polarity scores, extracted by sentiment analysis, in order to detect potential anomalies. It exploits the taxonomies and the techniques presented in this paper (Sections 4 and 5).

In our scenario, the urban area is modeled as an MIoT consisting of a set of IoTs $\{\mathcal{I}_1, \mathcal{I}_2, \cdots, \mathcal{I}_m\}$, each one associated with a portion of the area. The set of the objects of $\mathcal{M}$ comprises both the set of sensors, installed in the various smart lamps, and the set of personal devices of people who are moving around them. If an object o_j of the MIoT is active in the k^{th} portion of the urban area, it has an instance ι_{j_k} in the IoT $\mathcal{I}_k$. Clearly, when a person with a smart device o_j moves around different portions of the urban area, each one corresponding to a single IoT, o_j will have different instances, one for each IoT. An object o_j corresponding to a smart lamp sensor in the k^{th} urban area is fixed, and will contain only one instance ι_{j_k} in the corresponding IoT $\mathcal{I}_k$.

A transaction $Tr_{jq_{k_i}}$ between two object instances ι_{j_k} and ι_{q_k} can be generated in different ways. First of all, when citizens move around the various IoTs, they generate posts and alerts with their mobile devices. In this case, the transaction is associated with each post or alert. Sensors send transactions to the platform for sensed data, and smart lamps communicate with each other for parameter adjustments. Each of these events is translated into a transaction $Tr_{jq_{k_z}}$. Even the data mining module may send messages to the various smart lamps, thus generating transactions $Tr_{jq_{k_z}}$ in the MIoT.

6.2 Experiments

In this section, we present the experiments carried out to evaluate the performance of our approach from several viewpoints. Specifically, in Sect. 6.2.1, we illustrate our testbed. In Sect. 6.2.2, we analyze the forward problem from different perspectives. Finally, in Sect. 6.2.3, we focus on the experiments concerning the inverse problem.

6.2.1 Description of the testbed

To perform this analysis, we considered a reference scenario related to a smart city context. To model it, and to test our approach, we constructed a prototype. Furthermore, we realized an MIoT simulator.

In order to make "concrete" and "plausible" the simulated MIoT, our simulator needs to generate MIoTs having the characteristics specified by the user, whilst being as close as possible to real-world scenarios. In the simulator design, and in the construction of the MIoT used in the experiments, we followed the guidelines outlined in [33, 12, 13], where the authors highlight that one of the main factors used to build links in an IoT is node proximity.

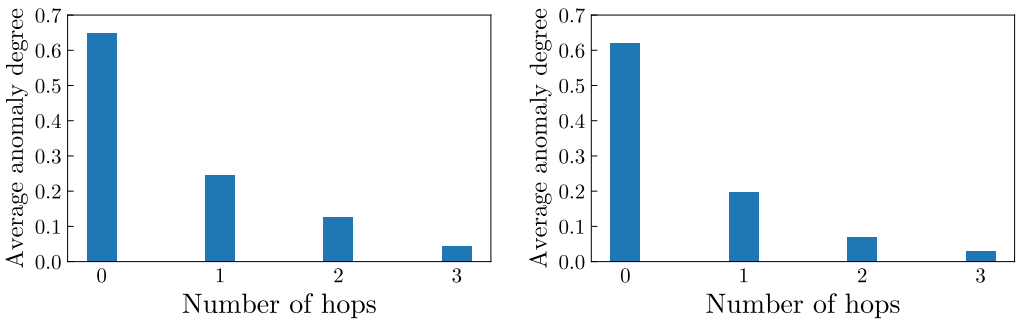
In order to reproduce the creation of transactions among objects, we decided to leverage information about a simulated smart city context. As for a dataset containing real-life paths in a smart city, we selected the one reported in `http://www.geolink.pt/ecmlpkdd2015-challenge/dataset.html`. This regards movements of objects, in terms of routes, in the city of Porto from July 1^{st} 2013 to June 30^{th} 2014. Each route contains several Points of Interest, corresponding to the GPS coordinates of each object as it moves in Porto. With this information at hand, our simulator associates an object (thus, creating a node) with one of the routes recorded in the dataset. Furthermore, it creates an arc between two nodes when the distance between the corresponding routes is less than a certain threshold th_d, for a predefined time interval th_t. The value of th_d and th_t can be specified through the constructor interface. Clearly, the higher is this value the more connected the constructed MIoT will be. When we defined the distribution of the transactions among the nodes, we leveraged scientific literature and used the corresponding results to properly tune our simulator. In particular, we adopted the values reported in [32].

The interested reader can find the MIoT created by our simulator for the experiments described in this paper at the Web address `http://daisy.dii.univpm.it/miot/datasets/anomaly-detection`. It consists of 1,256 nodes and six IoTs having 128, 362, 224, 280, 98 and 164 nodes, respectively. The constructed MIoT is returned in a format that can be directly processed by the cypher-shell of Neo4J. Some statistics about our dataset are reported in Table 2.

Parameter	*Value*
Number of nodes	1,256
Number of relationships	6,860
Mean outdegree	5.44
Mean indegree	5.58

Table 2: Parameter values for our simulator

We carried out all the tests presented in this section on a server equipped with an Intel I7 Quad



(a) (b)

Figure 1: Values of δ_{j_k} (corresponding to 0 hops) and average values of the anomaly degrees of all the nodes of $\mathcal{I}_k$ (on the left) and of the MIoT (on the right) being 1, 2 and 3 hops far from n_{j_k} in case of Presence-Hard-Contact anomalies

Core 7700 HQ processor and 16 GB of RAM, with the Ubuntu 16.04 operating system. To implement our approach, we adopted Python, as programming language, and Neo4J (Version 3.4.5), as underlying DBMS.

6.2.2 Analysis of the forward problem

Let us preliminarily define the concept of "number of hops" h_{jq_k} between the node n_{j_k} and another node n_{q_k} as the minimum number of arcs of the MIoT that must be traversed in order to reach n_{q_k} from n_{j_k}.

In a first step we analyzed the effects that the anomalous behavior of an object o_j had on the nodes of an MIoT. As pointed out in Sect. 5.1, given a node n_{j_k} of the IoT $\mathcal{I}_k$, its anomaly degree is represented by the parameter δ_{j_k}. This anomaly may propagate through the MIoT, thus affecting other nodes. To investigate this propagation, given an anomalous instance of an object o_j and the IoT $\mathcal{I}_k$, we measured the anomaly degree δ_{j_k} of n_{j_k} and the average of the anomaly degrees δ_{q_k} of all the nodes n_{q_k}, grouped by the number of hops from n_{j_k} to n_{q_k}. Moreover, we computed the same values but averaged through the IoT belonging to the MIoT. The same test has been run over 100 randomly chosen nodes, and results have been averaged over the runs.

Figure 1 shows the results obtained for Presence-Hard-Contact anomalies, while Figure 2 presents those regarding Presence-Soft-Contact anomalies. From the analysis of these figures it is possible to observe that the effects of an anomaly on a node spread over the surrounding nodes, even if they rapidly decrease against the number of hops. The corresponding trend follows a power law distribution. If we compare the left and the right distributions of Figures 1 and 2, we can observe that anomalies propagate more slowly on an MIoT than on a single IoT. However, this difference is negligible. Furthermore, there are no significant differences between Presence-Hard-Contact anomalies and Presence-Soft-Contact anomalies, except that the latter ones are slightly smaller than the former ones. This trend can be justified by considering that Presence-Soft-Contact anomalies are more difficult to be observed than Presence-Hard-Contact ones, since the former ones are not only required to show values higher (resp.,

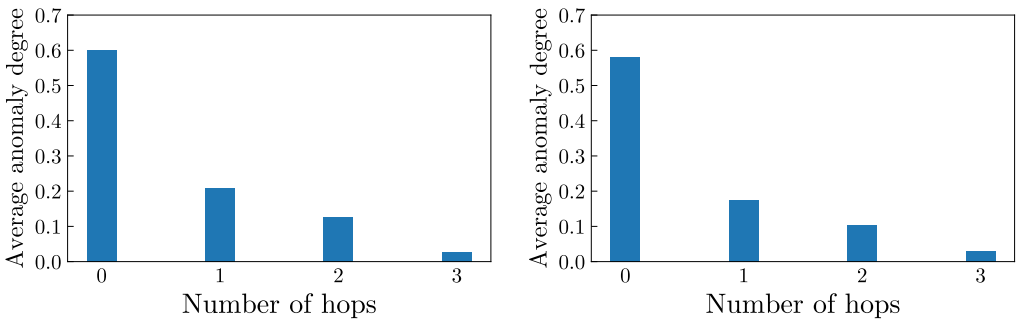


Figure 2: Values of δ_{j_k} (corresponding to 0 hops) and average values of the anomaly degrees of all the nodes of $\mathcal{I}_k$ (on the left) and of the MIoT (on the right) being 1, 2 and 3 hops far from n_{j_k} in case of Presence-Soft-Contact anomalies

lower) than a given threshold, but should also exhibit a trend that is monotonically increasing (resp., decreasing), within the time interval of interest. As the trends are very similar, in the following tests we focus only on Presence-Hard-Contact anomalies, without loss of generality.

Next, we investigated the effects that the anomaly of an object has on the other objects connected to it. In particular, given an object o_q, whose instances belong to the $ONbh$ of the instances of an anomalous object o_j in at least one IoT of the MIoT, we computed the value and the standard deviation[4] of δ_j and δ_q. We repeated this task 100 times with different pairs of objects o_j and o_q. Then, we averaged the values obtained over the runs. The corresponding results are shown in Figure 3, under the category ALL. As we can observe, the standard deviation of δ_j is very low. This result can be explained by the fact that all the instances of the anomalous object o_j present anomalies and, consequently, the corresponding anomaly degrees are almost uniform. By contrast, the value of δ_q is lower than the one of δ_j, exhibiting a very high standard deviation. This is explained by observing that the instances of o_q are not in the neighborhoods of the instances of o_j in all the IoTs of the MIoT. In fact, in some of them, they can be 2, 3 or more hops away from the instances of o_j. In some cases, they may even be disconnected from the instances of o_j.

As a next step, we repeated the previous experiment, enforcing some extra constraints, which defined three different scenarios. In the first (resp., second, third) one, all the instances of o_q were 1 (resp., 2, more than 2) hop(s) far from the instances of o_j; the third scenario includes also instances of o_q not connected to instances of o_j. The results obtained are shown in Figure 3 under the labels S_1, S_2 and S_3, respectively. Looking at the data labelled as ALL, these results are coherent with both the ones of Figure 1 and the ones of Figure 3. We can see that the effects of a single anomaly are rapidly reduced as soon as we move away from its origin. Furthermore, this experiment confirms what we pointed out in Section 5.1, i.e., that the anomaly degree δ is a parameter that really helps detecting the object that has caused the anomaly in the first place.

At this point, we investigated the number of nodes in an MIoT that turn out to be anomalous as a consequence of a single anomaly of an object o_j. Again, we repeated this experiment 100 times.

[4]Recall that δ_j and δ_q are computed by averaging the anomaly degrees of all the instances of o_j and o_q.

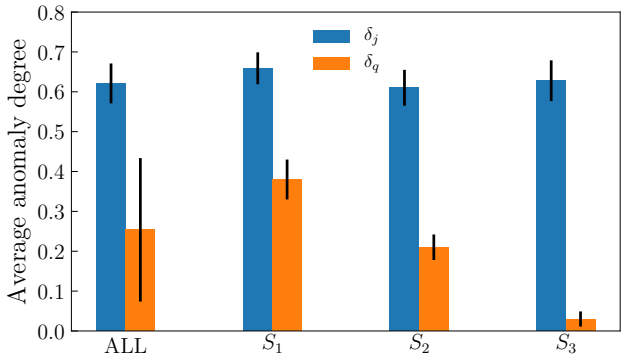


Figure 3: Anomaly degrees and the corresponding standard deviations in different scenarios

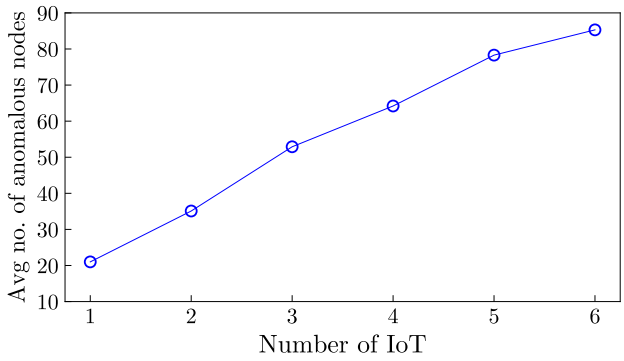


Figure 4: Average number of nodes affected by anomalies against the number of IoT which an anomalous object participates to

Each time, we selected an anomalous object of the MIoT. The selected objects had different number of instances in the MIoT, ranging from 1 to 6. For each run, we computed the number of anomalous nodes detected in the MIoT. Then, we computed the averages, by grouping the cases based on the number of instances of the anomalous objects and, therefore, based on the number of IoTs of the MIoT involved in the anomaly.

The results obtained are shown in Figure 4, which shows how the number of anomalous nodes increases against the number of IoTs in a roughly linear way. This trend can be explained by considering that, even when the number of objects having instances in many IoTs is usually limited with respect to the number of objects having instances in few IoTs, their anomalous behavior affects numerous nodes across several IoT and, consequently, their effect is amplified. On the contrary, anomalies observed on an object having instances in only one or two IoTs are more frequent. Yet, this is counterbalanced by the fact that each of these nodes only exerts a limited and localized impact, which affects only few nodes.

Then, we aimed to characterize which of the node properties impacted the spread of anomalies the most. We repeated the previous experiment; but instead of choosing anomalous nodes randomly, we selected them based on their characteristics. A first characteristic that we considered was the

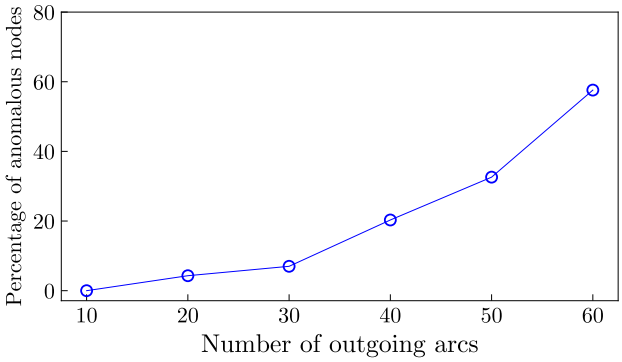


Figure 5: Average percentage of anomalous nodes against their degree centrality

outdegree of a node, i.e., the number of its outgoing arcs. In the various runs, we selected nodes with different outdegrees ranging from 10 to 60. For each of these values, we measured the average number of anomalous nodes throughout the MIoT detected by our approach. The results are illustrated in Figure 5, which clearly shows that the outdegree of anomalous nodes has a significant impact on the spread of the anomaly over the network. This result was not surprising, since it is consistent with the results about the information diffusion in social network analysis [62].

However, we argue that there is another form of centrality in social network analysis, which could be very promising as a node property to impact the spread of anomalies. This measure is closeness centrality. We recall that the closeness centrality of a node is defined as the reciprocal of the sum of the lengths of the shortest paths between the node itself and all the other nodes of the network.

Thus, we repeated the previous experiment; but this time we selected the anomalous nodes based on their closeness centrality. The values of this parameter for the nodes selected ranged from 0.05 to 0.45. The results obtained are shown in Figure 6, where we can observe that our intuition was right. Closeness centrality is really a key parameter in the spread of anomalies in an MIoT. It is even more important than degree centrality in this task. In our opinion, this result is extremely interesting because the impact of closeness centrality on anomaly diffusion is substantial, whilst the role of this parameter was a-priori much less obvious than the one of degree centrality.

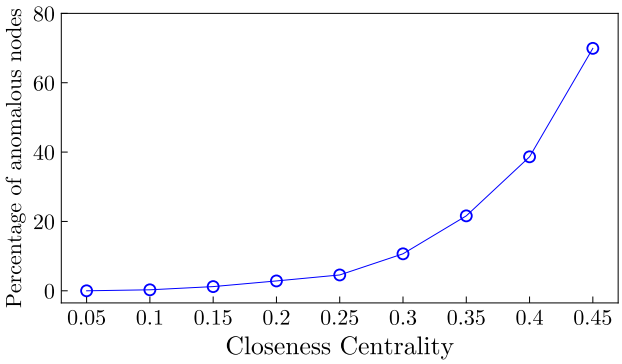


Figure 6: Average percentage of anomalous nodes against their closeness centrality

As a final test on the forward problem, we evaluated the running time necessary to compute the anomaly degree δ_j of an object o_j in an MIoT against the number of its nodes. The results obtained are reported in Figure 7, where we can observe a polynomial (specifically, a quadratic) dependency of the running time against the number of nodes of the MIoT. This can be explained by the fact that, during the computation of the recursive formula of δ_{j_k}, the values of α_{jq_k} tend to 0 rapidly while moving away from the node n_{j_k}.

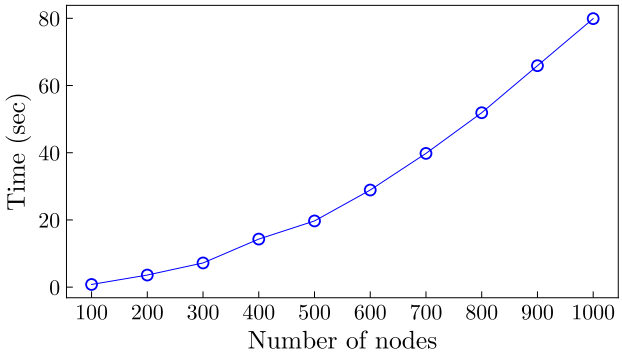


Figure 7: Running time (in seconds) needed to compute δ_j in an MIoT against the number of its nodes

6.2.3 Analysis of the inverse problem

In this section, we present the results of the tests we carried out to validate our approach for solving the inverse problem. We recall that our solution to this problem starts from an i-arc of an MIoT that presents an anomaly whose origin is not known. It applies a greedy algorithm, which aims at detecting the node that originated the anomaly.

During this test, we repeated 100 times the following tasks. We simulated an anomaly on an object and, then, we randomly selected an anomalous i-arc from the whole MIoT. We applied our solution of the inverse problem on this arc and computed the following:

- the number of hits, i.e., the percentage of times our approach detected the anomaly source correctly (we call S_0 this scenario);
- the percentage of times our approach terminated in a node belonging to the $ONbh$ of the anomalous node and, therefore, being 1 hop away from it (we call S_1 this scenario);
- the percentage of times our approach terminated in a node being 2 hops far from the anomalous node (we call S_2 this scenario);
- the percentage of times our approach terminated in a node being more than 2 hops away from the anomalous node (we call S_3 this scenario).

The results obtained are reported in Figure 8. They show that our approach is capable of correctly identifying the anomaly source in most cases. In a fraction of cases it stops very near to the anomalous

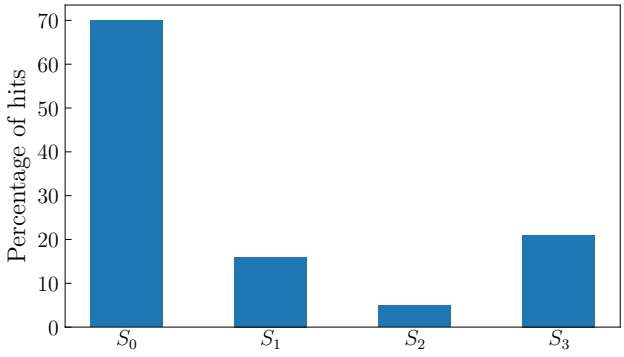


Figure 8: Percentage of times when our approach correctly detects the anomaly source (indicated by the label 0) or terminates in a node being 1, 2 or more than 2 hops far from it

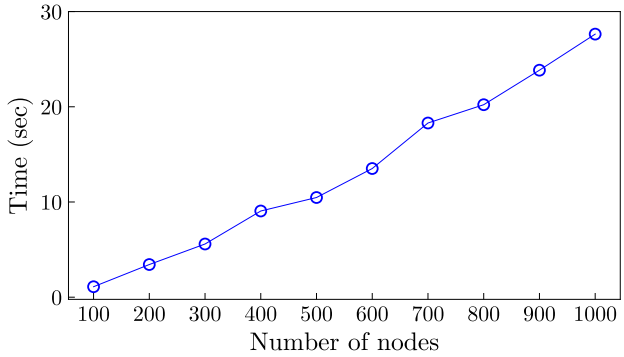


Figure 9: Average running time (in seconds) of our approach for solving the inverse problem

node, i.e., 1 or 2 hops away from it. The slightly higher frequency of the fourth case can be explained by the fact that the starting i-arc of the test is chosen randomly and, therefore, can be very far from the anomalous node. As a consequence, it comprises a relatively high number of cases (3, 4, 5 or more hops away from the anomalous object).

Next, we computed the average running time of our approach. Similarly to what we have done for the forward problem, we evaluated this time against the number of the MIoT nodes. The results obtained are shown in Figure 9, where we can observe that the running time increases polynomially against the number of MIoT nodes. This result can be explained by the fact that the greedy algorithm underlying our approach reaches the correct node, or a near one, in few iterations and by the fact that, on average, an anomaly on an i-arc can be observed only when this is not too far away from the node where the anomaly originated.

7 Conclusion

In this paper, we have presented a first attempt to investigate and classify anomalies in an MIoT. Our proposal consists of two main components. The first one is a new methodological framework that can

make future investigations in this research field easier, more coherent and more uniform. Indeed, our framework extends existing methods to the case of anomaly detection in an MIoT, whilst also allowing the definition of new cases. Another important contribution is the extension to the anomaly detection in MIoT of the so-called forward problem and inverse problem, which have been largely investigated and employed in scientific literature but were never analyzed in this research field. We also introduced a use case on a smart lighting system for an MIoT deployed in a smart city.

Our experiments have provided interesting outcomes about the capability of detecting anomalies and their effects in an MIoT. For instance, they revealed that: *(i)* the effects of an anomaly on a node spread over the surrounding nodes, even though they rapidly decrease against the distance; *(ii)* the anomaly degree δ defined in this paper is a parameter that really helps the detection of the anomalous object in a network; *(iii)* the number of nodes affected by an anomaly increases against the number of IoTs in a roughly linear way; *(iv)* degree centrality and, even more, closeness centrality are really key parameters in the spread of anomalies in an MIoT.

In the future, we can foresee several developments of this research. First of all, we would like to extend our framework to social networking and/or social internetworking scenarios, where humans and objects simultaneously inter-operate. In fact, the investigation of mixed networks, consisting of humans and smart/social objects, is attracting increasing interest among researchers. Next, we plan to extend our studies on MIoT anomalies for predictive maintenance, in such a way as to optimize the maintenance of production lines. Last, but not least, we think that several results obtained for MIoTs can be further exploited by applying some sort of "feedback", to identify new topics and new approaches for the investigation of human behavior in Online Social Networks.

Acknowledgments

This work was partially supported by: *(i)* the Italian Ministry for Economic Development (MISE) under the project "Smarter Solutions in the Big Data World", funded within the call "HORIZON2020" PON I&C 2014-2020 (CUP B28I17000250008), *(ii)* the Department of Information Engineering at the Polytechnic University of Marche under the project "A network-based approach to uniformly extract knowledge and support decision making in heterogeneous application contexts" (RSAB 2018), and *(iii)* the PRIN Project FluidWare funded by Italian Government (MIUR) N. 2017KRC7KT.

References

[1] M. Abulaish, A. Kamal, and M.J. Zaki. A Survey of Figurative Language and Its Computational Detection in Online Social Networks. *ACM Transaction on the Web*, 14(1):3:1–3:52, 2020. ACM.

[2] M. Ahmed. Collective anomaly detection techniques for network traffic analysis. *Annals of Data Science*, 5(4):497–512, 2018. Springer.

[3] M. Ahmed and A.N. Mahmood. Novel approach for network traffic pattern analysis using clustering-based collective anomaly detection. *Annals of Data Science*, 2(1):111–130, 2015. Springer.

[4] M. Ahmed, A.N. Mahmood, and J. Hu. A survey of network anomaly detection techniques. *Journal of Network and Computer Applications*, 60:19–31, 2016. Elsevier.

[5] M. Ahmed and A.S.S.M. Barkat Ullah. Infrequent pattern mining in smart healthcare environment using data summarization. *The Journal of Supercomputing*, 74(10):5041–5059, 2018. Springer.

[6] L. Akoglu, M. McGlohon, and C. Faloutsos. Oddball: Spotting anomalies in weighted graphs. In *Proc. of the Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining, (PAKDD'10) Part II*, pages 410–421, Hyderabad, India, 2010. Lecture Notes in Computer Science, Springer.

[7] L. Akoglu, H. Tong, and D. Koutra. Graph based anomaly detection and description: a survey. *Data Mining and Knowledge Discovery*, 29(3):626–688, 2015. Springer.

[8] S.A. Aljawarneh and R. Vangipuram. Garuda: Gaussian dissimilarity measure for feature representation and anomaly detection in internet of things. *The Journal of Supercomputing*, (11227):1–38, 2018. Springer US.

[9] F. Amato, V. Moscato, A. Picariello, and F. Piccialli. SOS: A multimedia recommender System for Online Social networks. *Future Generation Computer Systems*, 93:914–923, 2019. Elsevier.

[10] K. Anand, J. Kumar, and K. Anand. Anomaly detection in online social network: A survey. In *Proc. of the 2017 International Conference on Inventive Communication and Computational Technologies (ICICCT '17)*, pages 456–459, Coimbatore, India, 2017. IEEE.

[11] L. Atzori, A. Iera, and G. Morabito. SIoT: Giving a social structure to the Internet of Things. *IEEE Communications Letters*, 15(11):1193–1195, 2011. IEEE.

[12] L. Atzori, A. Iera, and G. Morabito. Understanding the Internet of Things: definition, potentials, and societal role of a fast evolving paradigm. *Ad Hoc Networks*, 56:122–140, 2017. Elsevier.

[13] L. Atzori, A. Iera, G. Morabito, and M. Nitti. The Social Internet of Things (SIoT)– when social networks meet the Internet of Things: Concept, architecture and network characterization. *Computer networks*, 56(16):3594–3608, 2012. Elsevier.

[14] U.A.B.U.A. Bakar, H. Ghayvat, S.F. Hasanm, and S.C. Mukhopadhyay. *Activity and Anomaly Detection in Smart Home: A Survey*, pages 191–220. Springer International Publishing, Cham, 2016.

[15] G. Baldassarre, P. Lo Giudice, L. Musarella, and D. Ursino. The MIoT paradigm: main features and an "ad-hoc" crawler. *Future Generation Computer Systems*, 92:29–42, 2019. Elsevier.

[16] M. Behniafar, A.R. Nowroozi, and H.R. Shahriari. A survey of anomaly detection approaches in internet of things. *The ISC International Journal of Information Security*, 10(2):79–92, 2018. Iranian Society of Cryptology.

[17] P.V. Bindu, P. Santhi Thilagam, and D. Ahuja. Discovering suspicious behavior in multilayer social networks. *Computers in Human Behavior*, 73:568–582, 2017. Elsevier.

[18] L. Bontemps, V.L. Cao, J. McDermott, and N. Le-Khac. Collective anomaly detection based on long short-term memory recurrent neural networks. In *Proc. of the International Conference on Future Data and Security Engineering (FDSE'16)*, pages 141–152, Can Tho City, Vietnam, 2016.

[19] F. Buccafurri, V.D. Foti, G. Lax, A. Nocera, and D. Ursino. Bridge Analysis in a Social Internetworking Scenario. *Information Sciences*, 224:1–18, 2013. Elsevier.

[20] D. Camacho, A. Panizo-LLedot, G. Bello-Orgaz, A. Gonzalez-Pardo, and E. Cambria. The four dimensions of social network analysis: An overview of research methods, applications, and software tools. *Information Fusion*, 63:88–120, 2020. Elsevier.

[21] U. Can and B. Alatas. A new direction in social network analysis: Online social network analysis problems and applications. *Physica A: Statistical Mechanics and its Applications*, 535:122372, 2019. Elsevier.

[22] F. Cauteruccio, L. Cinelli, G. Fortino, C. Savaglio, and G. Terracina. Using sentiment analysis and automated reasoning to boost smart lighting systems. In *Proc. of the 12th International Conference in Internet and Distributed Computing Systems (IDCS 2019)*, volume 11874 of *LNCS*, pages 69–78, Naples, Italy, 2019. Springer.

[23] F. Cauteruccio, G. Fortino, A. Guerrieri, A. Liotta, D.C. Mocanu, C. Perra, G. Terracina, and M.T. Vega. Short-long term anomaly detection in wireless sensor networks based on machine learning and multi-parameterized edit distance. *Information Fusion*, 52:13–30, 2019. Elsevier.

[24] V. Chandola, A. Banerjee, and V. Kumar. Anomaly detection: A survey. *ACM Computer Surveys*, 41(3):15:1–15:58, 2009. ACM.

[25] D. Chen, H. Gao, L. Lu, and T. Zhou. Identifying influential nodes in large-scale directed networks: the role of clustering. *PloS one*, 8(10):e77455, 2013. Public Library of Science.

[26] F. Chen and D.B. Neill. Non-parametric scan statistics for event detection and forecasting in heterogeneous social media graphs. In *Proc. of the International Conference on Knowledge Discovery and Data Mining (KDD'14)*, pages 1166–1175, New York, NY, USA, 2014. ACM.

[27] Z. Chen, W. Hendrix, and N.F. Samatova. Community-based anomaly detection in evolutionary networks. *Journal of Intelligent Information Systems*, 39(1):59–85, 2012. Springer.

[28] S. Fakhraei, J.R. Foulds, M.V.S. Shashanka, and L. Getoor. Collective Spammer Detection in Evolving Multi-Relational Social Networks. In *Proc. of the International Conference on Knowledge Discovery and Data Mining (KDD'15)*, pages 1769–1778, Sydney, Australia, 2015. ACM.

[29] P. Garcia-Teodoro, J.E. Díaz-Verdejo, G. Maciá-Fernández, and E. Vázquez. Anomaly-based network intrusion detection: Techniques, systems and challenges. *Computers & Security*, 28(1-2):18–28, 2009. Elsevier.

[30] S. Garg, K. Kaur, S. Batra, G. Kaddoum, N. Kumar, and A. Boukerche. A multi-stage anomaly detection scheme for augmenting the security in iot-enabled applications. *Future Generation Computer Systems*, 104:105–118, 2020. Elsevier.

[31] P. Gogoi, D.K. Bhattacharyya, B. Borah, and J. K. Kalita. A survey of outlier detection methods in network anomaly identification. *The Computer Journal*, 54(4):570–588, 2011. Elsevier.

[32] Peerless Research Group. Sensors in Distribution: On the Cusp of New Performance Efficiencies. *https://www.logisticsmgmt.com/wp_content/honeywell_wp_sensors_022316b.pdf*, 2015.

[33] I.D. Guedalia, J. Guedalia, R.P. Chandhok, and S. Glickfield. Methods to discover, configure, and leverage relationships in Internet of Things (IoT) networks, feb 20 2018. US Patent 9,900,171.

[34] R. K. Gunupudi, M. Nimmala, N. Gugulothu, and S. R. Gali. Clapp: A self constructing feature clustering approach for anomaly detection. *Future Generation Computer Systems*, 74:417–429, 2017. Elsevier.

[35] D.M. Hawkins. *Identification of outliers / D.M. Hawkins*. Chapman and Hall London ; New York, New York, 1980.

[36] A. Høst-Madsen and J. Zhang. Coding of graphs with application to graph anomaly detection. *CoRR*, abs/1804.02469, 2018.

[37] K. Hundman, V. Constantinou, C. Laporte, I. Colwell, and T. Söderström. Detecting Spacecraft Anomalies Using LSTMs and Nonparametric Dynamic Thresholding. In *Proc. of the International Conference on Knowledge Discovery and Data Mining (KDD'18)*, pages 387–395, London, UK, 2018. ACM.

[38] L. Jain and R. Katarya. Discover opinion leader in online social network using firefly algorithm. *Expert Systems with Applications*, 122:1–15, 2019. Elsevier.

[39] V. Jyothsna and V.V. Rama Prasad. Article: A review of anomaly based intrusion detection systems. *International Journal of Computer Applications*, 28(7):26–35, 2011. IJCA Journal.

[40] H. Kim, R. Çetin Atalay, and E. Gelenbe. G-Network Modelling Based Abnormal Pathway Detection in Gene Regulatory Networks. In *Proc. of the International Symposium on Computer and Information Sciences (ISCIS'11)*, pages 257–263, London, UK, 2011.

[41] H. Kim and E. Gelenbe. Anomaly detection in gene expression via stochastic models of gene regulatory networks. *BMC Genomics*, 10(3):S26, 2009. BMC Genomics.

[42] J. Kim and M. Hastak. Social network analysis: Characteristics of online social networks after a disaster. *International Journal of Information Management*, 38(1):86–96, 2018.

[43] E.L. Koua, A.M. MacEachren, and M. Kraak. Evaluating the usability of visualization methods in an exploratory geovisualization environment. *International Journal of Geographical Information Science*, 20(4):425–448, 2006. Taylor & Francis.

[44] W. Li, S. Tug, W. Meng, and Y. Wang. Designing collaborative blockchained signature-based intrusion detection in iot environments. *Future Generation Computer Systems*, 96:481–489, 2019. Elsevier.

[45] J. Lin, E.J. Keogh, A.W. Fu, and H. Van Herle. Approximations to magic: Finding unusual medical time series. In *Proc. of the 18th IEEE Symposium on Computer-Based Medical Systems (CBMS 2005), 23-24 June 2005, Dublin, Ireland*, pages 329–334, 2005. IEEE Computer Society.

[46] G. Marra, F. Ricca, G. Terracina, and D. Ursino. Information Diffusion in a Multi-Social-Network Scenario: A framework and an ASP-based analysis. *Knowledge and Information Systems*, 48(3):619–648, 2016. Springer.

[47] B.A. Miller, N. Arcolano, and N. T. Bliss. Efficient anomaly detection in dynamic, attributed graphs: Emerging phenomena and big data. In *Proc. of the 2013 IEEE International Conference on Intelligence and Security Informatics, Seattle, WA, USA, June 4-7, 2013*, pages 179–184, 2013. IEEE.

[48] R. Navigli and S.P. Ponzetto. BabelNet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, 193:217–250, 2012. Elsevier.

[49] N. Nesa, T. Ghosh, and I. Banerjee. Non-parametric sequence-based learning approach for outlier detection in iot. *Future Generation Computer Systems*, 82:412–421, 2018. Elsevier.

[50] T.V. Nguyen, N.T. Tran, and S. Le Thanh. An anomaly-based network intrusion detection system using deep learning. In *Proc. of the 2017 International Conference on System Science and Engineering (ICSSE)*, pages 210–214, Ho Chi Minh City, Vietnam, 2017. IEEE.

[51] L. Page, S. Brin, R. Motwani, and T. Winograd. The PageRank Citation Ranking: Bringing Order to the Web. In *Proc. of the Seventh International World-Wide Web Conference (WWW 1998)*, pages 161–172, Brisbane, Australia, 1998. Elsevier.

[52] H. Haddad Pajouh, R. Javidan, R. Khayami, D. Ali, and K.R. Choo. A two-layer dimension reduction and two-tier classification model for anomaly-based intrusion detection in iot backbone networks. *IEEE Transactions on Emerging Topics in Computing*, pages 1–1, 2019. IEEE.

[53] S.J. Russell and P. Norvig. *Artificial intelligence: a modern approach*. Malaysia; Pearson Education Limited,, 2016.

[54] S. Salvador, P. Chan, and J. Brodie. Learning states and rules for time series anomaly detection. In *Proc. of the Seventeenth International Florida Artificial Intelligence Research Society Conference, Miami Beach, Florida, USA*, pages 306–311, 2004. AAAI Press.

[55] D. Savage, X. Zhang, X. Yu, P. Chou, and Q. Wang. Anomaly detection in online social networks. *Social Networks*, 39:62–70, 2014. Elsevier.

[56] M. Shao, J. Li, F. Chen, H. Huang, S. Zhang, and X. Chen. An efficient approach to event detection and forecasting in dynamic multivariate social media networks. In *Proc. of the 26th International Conference on World Wide Web*, pages 1631–1639, Perth, Australia, 2017. ACM.

[57] V. Sharma, I. You, and R. Kumar. Isma: Intelligent sensing model for anomalies detection in cross platform osns with a case study on iot. *IEEE Access*, 5:3284–3301, 2017. IEEE.

[58] S. Shekhar, C. Lu, and P. Zhang. Detecting graph-based spatial outliers: algorithms and applications (a summary of results). In *Proc. of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining, San Francisco, CA, USA, August 26-29, 2001*, pages 371–376, 2001. ACM.

[59] N. Shrivastava, A. Majumder, and R. Rastogi. Mining (social) network graphs to detect random link attacks. In *Proc. of the 24th International Conference on Data Engineering, ICDE 2008, April 7-12, 2008, Cancún, Mexico*, pages 486–495, 2008. IEEE Computer Society.

[60] D. Simon. *Evolutionary optimization algorithms*. John Wiley & Sons, 2013.

[61] S. Sudrich, J. De Melo Borges, and M. Beigl. Anomaly detection in evolving heterogeneous graphs. In *Proc. of the International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData)*, pages 1147–1149, Exeter, UK, 2017. IEEE Computer Society.

[62] M. Tsvetovat and A. Kouznetsov. *Social Network Analysis for Startups: Finding connections on the social web*. Sebastopol, CA, USA, 2011. O'Reilly Media, Inc.

[63] J.M. Vanerio and P Casas. Ensemble-learning approaches for network security and anomaly detection. In *Proc. of the Workshop on Big Data Analytics and Machine Learning for Data Communication Networks, Big-DAMA@SIGCOMM 2017*, pages 1–6, Los Angeles, CA, USA, 2017. ACM.

[64] A. M. Vegni, V. Loscri, and A. Benslimane. SOLVER: A Framework for the Integration of Online Social Networks with Vehicular Social Networks. *IEEE Network*, 34(1):204–213, 2020. IEEE.

[65] F. Wang, Z. Wang, Z. Li, and J.R. Wen. Concept-based Short Text Classification and Ranking. In *Proc. of the International Conference on Information and Knowledge Management (CIKM'14)*, pages 1069–1078, Shangai, China, 2014. ACM.

[66] T. Wang, G. Zhang, A. Liu, M. Z. A. Bhuiyan, and Q. Jin. A secure iot service architecture with an efficient balance dynamics based on cloud and edge computing. *IEEE Internet of Things Journal*, 6(3):4831–4843, 2019. IEEE.

[67] W. Yu, J. Li, M. Z. A. Bhuiyan, R. Zhang, and J. Huai. Ring: Real-time emerging anomaly monitoring system over text streams. *IEEE Transactions on Big Data*, 5(4):506–519, 2019. IEEE.

[68] Y. Yu, H. Yan, H. Guan, and H. Zhou. Deephttp: Semantics-structure model with attention for anomalous http traffic detection and pattern mining. *CoRR*, abs/1810.12751, 2018. IEEE.

[69] B.B. Zarpelão, R.S. Miani, C.T. Kawakani, and S.C. de Alvarenga. A survey of intrusion detection in internet of things. *Journal of Network and Computer Applications*, 84:25–37, 2017. Elsevier.