Ad Hoc Networks 35 (2015) 14–36

Contents lists available at ScienceDirect

Ad Hoc Networks

journal homepage: www.elsevier.com/locate/adhoc

Ensembles of incremental learners to detect anomalies in ad hoc sensor networks

Hedde H.W.J. Bosman [a,b,*], Giovanni Iacca [a], Arturo Tejada [a], Heinrich J. Wörtche [a], Antonio Liotta [b]

[a] INCAS3, Dr. Nassaulaan 9, 9401 HJ Assen, The Netherlands
[b] Department of Electrical Engineering, Eindhoven University of Technology, P.O. Box 513, 5600MB Eindhoven, The Netherlands

ARTICLE INFO

Article history:
Received 13 December 2014
Revised 9 July 2015
Accepted 17 July 2015
Available online 26 July 2015

Keywords:
Anomaly detection
Wireless sensor networks
Online learning
Incremental learning
Ensemble methods

ABSTRACT

In the past decade, rapid technological advances in the fields of electronics and telecommunications have given rise to versatile, ubiquitous decentralized embedded sensor systems with ad hoc wireless networking capabilities. Typically these systems are used to gather large amounts of data, while the detection of anomalies (such as system failures, intrusion, or unanticipated behavior of the environment) in the data (or other types or processing) is performed in centralized computer systems. In spite of the great interest that it attracts, the systematic porting and analysis of centralized anomaly detection algorithms to a decentralized paradigm (compatible with the aforementioned sensor systems) has not been thoroughly addressed in the literature. We approach this task from a new angle, assessing the viability of localized (in-node) anomaly detection based on machine learning. The main challenges we address are: (1) deploying decentralized, automated, online learning, anomaly detection algorithms within the stringent constraints of typical embedded systems; and (2) evaluating the performance of such algorithms and comparing them with that of centralized ones. To this end, we first analyze (and port) single and multi-dimensional input classifiers that are trained incrementally online and whose computational requirements are compatible with the limitations of embedded platforms. Next, we combine multiple classifiers in a single online ensemble. Then, using both synthetic and real-world datasets from different application domains, we extensively evaluate the anomaly detection performance of our algorithms and ensemble, in terms of precision and recall, and compare it to that of well-known offline, centralized machine learning algorithms. Our results show that the ensemble performs better than each individual decentralized classifier and that it can match the performance of the offline alternatives, thus showing that our approach is a viable solution to detect anomalies, even in environments with little a priori knowledge.

1. Introduction

The Internet of Things (IoT), a decade-long vision of a seamless networking where classic networked systems coexist with ubiquitous devices, is becoming reality [1,2]. Nowadays, not only computers, tablets, and smart phones, but also vehicles, white goods, and other Internet-enabled industrial and domestic apparatuses can be connected into a single, heterogeneous world-wide network. This opens up many possibilities for context-aware applications, such as smart buildings, smart cities, and autonomous distributed systems, among others. Within this context, it is of great interest in many IoT applications to embed in the network the ability of

* Corresponding author at: INCAS[3], P.O. Box 797, 9400AT Assen, The Netherlands. Tel.: +31 592 860 000.
E-mail address: heddebosman@incas3.eu, sgorpi@gmail.com (H.H.W.J. Bosman).

http://dx.doi.org/10.1016/j.adhoc.2015.07.013
1570-8705/

detecting, in an online, and decentralized fashion, anomalies in the sensed or received information. This is the focus of this paper.

One of the first classes of IoT devices are the wireless sensor networks (WSN). These consist of a set of nodes, which are (generally) resource-limited embedded platforms endowed with sensors. Such nodes construct an ad hoc network to communicate with each other and with one or more sink nodes (i.e., nodes connected to a central facility for data storage and analysis). For over a decade, the community around WSN has focused mainly on the optimization of resource usage, e.g., network protocol design. Recently, however, the community's focus is shifting to the applications of WSN. Typical applications can be found in agriculture, where WSN are used to provide detailed insight on soil conditions [3], or in environmental monitoring, where they are used, for instance, to measure the effect of global warming on glaciers [4]. Other application domains include civil engineering (with various successful case studies in infrastructural monitoring [5], optimal tunnel lighting conditions control [6], water distribution network monitoring [7]), and health care (with many applications such as the monitoring of falls, medical intake or medical condition [8]). Lately, WSN are slowly being adopted also in industrial settings [9], although these applications are tightly controlled due to stringent reliability, safety and security requirements.

In many such applications, as well as in other IoT scenarios, the most compelling challenge is often to analyze, possibly in real-time, the big datasets that are generated. For domain specialists, such an analysis could provide new inferred knowledge about the sensed environment/processes, that could in turn be used to, for instance, improve their modeling. However, to make this analysis possible, automated analysis strategies [10] and big data analysis techniques [11] are needed, since such large datasets cannot be processed manually.

One special case of data analysis is the detection of *anomalies*, i.e., of specific events or patterns in the data that are unexpected [12]. Although the generic notion of an anomaly is rather intuitive, the specific cause (and nature) of such events varies widely across application domains. For example, in environmental monitoring an anomaly can be due to a sensor fault or an (unpredicted) environmental event [13], while in network intrusion detection, an anomaly can be the result of an intruder (often malicious) in the network [14]. Nevertheless, several generic anomaly detection methods [15] have been designed to find patterns in the data and recognized when they are unexpectedly broken. Depending on the quantity (and quality) of a priori information available about the environment or process at hand, these methods include a combination of formal techniques, rules, or data mining techniques from computational intelligence [16], pattern recognition and machine learning [17,18]. One of the limitations of these techniques, though, is that they often make use of statistical data models (possibly inferred by unsupervised learning) or detection rules that are not always available under limited information conditions. Furthermore, these techniques generally require large amounts of data to be available and stored in memory, and considerable computer processing power. Therefore, anomaly detection based on these methods is typically performed in large, centralized (data mining) computing systems.

Such methods are clearly not compatible with IoT applications for several reasons. On the one hand, there is no natural centralized processing location for the continuously growing number of IoT devices connected to the Internet. Even if one of these (or one powerful computer) were selected as the centralized processing node, the need to transport data from all devices to this node would quickly overwhelm the network communication capacity and increase its response time. On the other hand, most IoT devices are expected to have limited power and computing resources (e.g., memory), so they could hardly take the place of a central computing node (specially in WSN applications). For these reasons, the need and interest for decentralized data processing (including anomaly detection) are steadily increasing (see, e.g., [19–21]). Decentralization can take place at the networking level (see, e.g., cognitive radio research [22,23]) or at the application level (e.g., probabilistic inference and regression [24–26], decision making via fuzzy logics [27] or voting techniques [28]).

In spite of these contributions, few attempts have been made to develop methods for *online* learning of models in networked embedded systems. Online learning is needed to provide networked systems (e.g., WSN) with the ability to adapt to local working conditions without using a priori information. This is specially important for distributed anomaly detection. Among the available methods, some require a hierarchical organization of the network, such as in the multi-level clustering methods proposed in [29,30]; while others are limited only to a specific class of anomalies [31], or make assumptions on the inter-node correlations and the underlying process under observation [32]. In any case, what is offered by the current literature is often based on a preliminary learning phase to build a model of the monitored system. Such approaches require several offline steps of adaptation to target a specific application domain. To the best of our knowledge, no general-purpose (w.r.t. applications and anomalies), online (i.e., with continuous learning), decentralized anomaly detection framework exists for WSNs.

In this paper, we fill this gap in two ways: (1) we present a general-purpose, online learning, decentralized anomaly detection framework that includes a heterogeneous set of local anomaly detection algorithms (applicable on a node either independently or in the form of an *ensemble*), and whose computational requirements are compatible with the stringent limitations of the embedded platforms typically used in WSNs. We build upon previous preliminary works around individual classifiers [33–35], elaborate on the challenges and choices of learning methodologies for limited-resource devices, and subject the methods to a significantly larger experimental campaign. (2) We evaluate the performance of such algorithms in contrast to centralized anomaly detection methods. For evaluation purposes, labeled datasets are required, which we obtained through synthetic generation and from real-world applications, developed in-house or available through public sources. Moreover, we review the evaluation methods, which are based on the confusion matrix metrics, describing how we account for false positives caused by anomalies in correlated sensors, and for false positives caused by a delayed detection. Through this broad