Anomaly Detection on IOT Data for Smart City

Pierfrancesco Bellini
DISIT Lab
University of Florence
Florence, Italy
pierfrancesco.bellini@unifi.it

Daniele Cenni
DISIT Lab
University of Florence
Florence, Italy
daniele.cenni@unifi.it

Paolo Nesi
DISIT Lab
University of Florence
Florence, Italy
paolo.nesi@unifi.it

Mirco Soderi
DISIT Lab
University of Florence
Florence, Italy
mirco.soderi@unifi.it

***Abstract*— Smart Cities are probably on the more complex environment for IOT data collection. IOT data could have different producers, sample rates, periodic and aperiodic, typical trends, structures and stacks, faults, etc. Thus, a strongly flexible and scalable solution is needed to avoid investing huge amount of resources in anomaly detection that has to be done in real time and has to be agnostic to the above-mentioned problems. This paper presents a solution for automatic detection of anomalies. The proposed approach scales seamlessly and integrates in different contexts, featuring different sensor types, protocols, and data formats, and computationally cheap. The research has been developed in the context of Snap4City PCP Select4Cities project and is presently implemented in the Https://www.snap4city.org solution adopted in several cities and regions.**

Keywords—anomaly detection, sentient cities, IoT, smart cities, machine learning

I. INTRODUCTION AND RELATED WORK

In recent years, there has been a substantial increase in the number of IoT devices in many sectors: smart city, industry, home, etc. With the increment of relevance of IOT data in process management, there is an increasing need to develop robust solutions for detecting anomalies which should be agnostic with respect to the data type and contexts. Problems in the ingested data are called anomalies in this paper. They can be a sign of serious problems in the upstream chain as well into the data ingestion process. For example, the early detection of a fault condition may lead to identify the needs of a maintenance intervention on a service, which may lead to save money, and in some cases the life. For example, when faults on data signal may lead to some incidental finding and thus may represent alarms of civil alerts conditions (for example, a problem on a water sensor would be due to the inception of hearth erosion as occurred in Lungarno Torrigiani in Florence, in 2018). In critical cases, there is the need to provide timely and accurate feedbacks, managing large amount of data. Massive Internet of Things (MIoT) is coming, and thus fast and reliable early warning solutions are needed [1], [2].

The anomalies can be divided in three categories: point, contextual and collective [3]. A **point** anomaly is nothing more than an outlier, i.e. a value significantly different in size than the rest of the data. A **contextual** anomaly (also called conditional anomaly [4]) is a value that is considered anomalous in a certain context, while it could be taken as normal in another, for example a time series measuring the number of cars on a road, at different hours of the day. **Collective** anomalies are not directly detectable by the observation of a limited number of samples, since they may emerge only from the observation of trends distributed over a sufficiently large period of time. Depending on the type of data, and the anomalies that are to be detected, different kinds of approaches have been used. For example, techniques using dissimilarity measures [5], percentiles [6], Gaussian Mixture models [7], Hierarchical Markov models [8], Bayesian models [9], Switching Hidden Semi-Markov models [22] have been applied. These are supervised approaches that require appropriately labelled data and are aimed at detecting **collective** anomalies. Machine learning approaches include the use of Support Vector Machines (SVM) [10], Multi-class SVM [11], Support Vectors [12], Convolutional Neural Networks and Recurrent Neural Networks [13]. Other techniques that have been applied in the IoT context include temporal and spatio-temporal frameworks, for example for analyzing air pollutants [14], Extreme Learning Machines [15], Clustering [16] and Multivariate Clustering [17]. In case of large datasets, where labelling data could be expensive, clustering techniques are of help. For example, for the case of networks intrusion detection, a clustering approach has been proposed for the detection of unseen cases [18]. Another proposed approach makes use of semi-supervised hierarchical stacking Temporal Convolutional Network (TCN), that deals with datasets with a limited number of labelled instances [19]. As a development of RNN, LSTM networks have been proposed to analyze time series data, and they proved to be very accurate in modelling long time series [20]. Through the use of LSTM Autoencoders, it is possible to reconstruct a time series and find out if it resembles a normal behavior by measuring the dissimilarity of the reconstructed signal from the original one. Variants to this schema have been proposed, for example that deal with multivariate time series (MTS) [21]. Many of the proposed approaches make use of limited datasets or address only one of the issues (e.g. they detect only one type of anomaly and are limited to a particular type of data).

This paper presents a solution for automatic detection of sensor anomalies in the context of IoT. The proposed approach scales seamlessly and integrates in different contexts, featuring different sensor types, protocols, and data formats. The aim of the present work is to provide a robust solution for fast and accurate anomaly detection at each level of the IoT stack, providing automatic alerts and valuable insights to the administrators of an IoT monitoring infrastructure. As the amount of data in IoT contexts often grows exponentially, there is a need to develop scalable solutions, which can be easily integrated with existing infrastructures, and which require low resource consumption. The objective of this research is to provide an intelligent fault detection system at every level of the IoT stack, so as to provide operators with valuable information on the status of systems that do not require large computing capabilities or waiting times incompatible with an IoT context.

The paper is organized as follows. Section II presents the requirements of an ideal anomaly detection for IOT also describing the most relevant problems. In Section III, the Snap34City infrastructure is presented. Section IV introduces a part of the data sets and samples. In Section V, the model

978-1-7281-6997-2/20/$31.00 ©2020 IEEE
DOI 10.1109/SMARTCOMP50058.2020.00087

for anomaly detection is presented, while Section VI presents the results. Conclusions are drawn in Section VII.

II. Requirements Analysis

According to the above state of the art analysis, the early detection of the anomalies on IoT data could be a valuable instrument for the detection of dysfunctions on the stack: monitored physical entities(home, water, pollutant, traffic, energy, etc.), IoT Devices, IoT Edge/Fog, IoT Brokers, authentication services, connections and network devices, servers, data shadow storage, indexes, and processes involved in the end-to-end chain of getting and showing the data. To this end, a number of requirements for a suitable tool for anomaly detection have been identified and are formalized in the following. So that, a perfect solution for anomaly detection should be capable to detect sensor related **anomalies** even if they are:

- **(structure)** belonging to IoT Devices with multiple sensors, and the single IoT Devices may belong to collection of IoT Devices. On this regard, the fault detected on a single sensor or on multiple sensors of an IoT Device could be a signal of fault detection at level of IoT Device. And similarly, for the IOT Device Collection or IoT Broker, Data Shadow storage, data index, etc.;
- **(moving)** located on IoT Devices that are moving, such as Mobile App, vehicles, air quality sensors located on busses, etc.;
- **(producer)** belonging to IoT sensors produced by different builders/producers, protocols, data formats, unit of measures, data types, sample rates, etc.;
- **(stack faults)** due to different causes/faults along the above described stack;
- **(noise)** affected by measurement noise, that has to be modelled as well. This means that the solution of anomaly detection should be resilient to the effect of noise;
- confined in one point **(outlier)** with respect to the typical bounds;
- contextual or collective **(conditional)**, which could depend on the context: time slot of the day, day of week, city area, government regulation, etc. This means that a description of context is also needed;
- **(typical trends)** referring to some classification and typical trends that are not met. For example, a value that is beyond the typical trend for Friday morning at 10:00, having as a reference the typical trends for each signal, that may emerge only from the observation of trends distributed over a sufficiently large period of time. This implies to take into account the context and seasonal trends; please note that trends can be similar while the same rate strongly different;
- **(period)** on signals that could be taken periodically or sporadically. In the context of the industry the acquisition of data is typically periodic due to the presence of a control system in which the sampling period is part of the mathematical model. On the other hand, in the context of Smart City and IoT in the other fields the data are not periodically sampled. It could be too expensive, and often sensors are programmed in a way such to avoid sending messages when the data value has not changed significantly. Thus, resulting in aperiodic signals, lacking a precise rhythm. This is also the reason for which each sampled sensor values should have a time stamp associated with and not demanding the time stamping to the IoT brokers, that could produce stamps affected by the latency of the network connection.
- **(rate)** on signals that could present different sampling rates. As described above, in the industrial field the sampling rate is part of the model, while in large solutions as smart city home automation, the rate is not constant. And a non-constant rate would not automatically lead to an anomaly.
- **(scalable)** on a huge amount of them. This means that the solution has to be scalable to avoid investing huge amount of resources in anomaly detection in real time.

If the solution is capable to detect anomalies according to the above requirements, this also means that it is capable to produce reliable results despite the adverse conditions of noise, mobility, change of period and rate, etc. In addition, the estimation of anomalies, combined with the knowledge of the above described end-2-end stack of data gathering and usage, can be used to understand in most cases, if the anomaly has been caused by a fault in the: (i) data chain rather than (ii) on physical environment, or (iii) on the device. It is therefore important to ensure not only the data quality at the source, through the implementation of processes that allow to monitor and possibly correct anomalies in the systems, but it is also of particular importance to be able to ensure the data quality after their processing. The quality of the models and services implemented after data processing is very much affected by the processes of aggregation, de-noising, and cleaning up of the data, which are necessary in an IoT context.

III. Snap4City overview

As mentioned in the introduction, the solution proposed in this paper has been developed in the context of Snap4City (https://www.snap4city.org). Snap4City provides services and data of several cities/Organizations such as: Firenze, Helsinki, Antwerp, Lonato del Garda, Santiago de Compostela, Pisa, Prato, Pistoia, Lucca, Arezzo, Grosseto, Livorno, Siena, Massa, Modena, Cagliari, Valencia, Pont du Gard, Dubrovnik, West Greek, Mostar; and from regions like Tuscany, Garda Lake, Sardegna, Belgium, Finland, Emilia Romagna, Spain, etc. Snap4City is open to your contributions, using Snap4City tools and contributing to its improvement, adding more tools and features, etc. Please join the community on this portal and on GitHub/disit. https://github.com/disit [23], [24], [25].

The service and data addressed area relate to the solution in place in Florence and Tuscany Region as a whole. In that area, Snap4City collects data coming from several different domains: mobility, environment, transport, health, mobile apps, etc. Snap4City is capable to keep under control the real time city evolution: reading sensors; computing and controlling key performance indicators (KPI); detecting unexpected evolutions; performing analytics; taking actions on strategies and alarms. Snap4City supports the city in the process of continuous control and supervision, tools for business intelligence, predictions, etc.

In the Snap4City platform, devices (sensors/actuators) are searchable within a web application called IoT Directory, developed for monitoring and managing the devices that are available in the city by their Context Brokers. The IoT Directory allows you to see IoT devices at the physical level and single sensors/actuators with their type (e.g., temperature, pression, velocity, humidity), and data type (e.g., float, integer). Depending on the Context Broker, a device is identified by a specific identification field or through the name of the channel/topic, according to which their values are published (e.g., in MQTT, NGSI and AMQP protocols). Snap4City features a huge number of device types related to air quality, bike/bus/ferry/tunnels traffic, car parking occupation, charging stations, noise level, pollution observations, smart bench observations, vehicle traffic, smart waste, weather.

IV. Data Analysis and Examples

In the context of the present work, data was crawled from air quality and traffic related devices. Tables I reports a set of sensor signal monitored. They are mainly air quality pollutant, weather conditions, and traffic flow data. Please note that, traffic flow data are typically periodic, daily and weekly, while not all the air quality data present some periodicity. The temperature and humidity data also have daily and seasonal periodicities, but not weekly. All of them may have relevant aperiodicity for sporadic events (traffic accidents, rains, hurricanes, etc., that may be on physical world, natural and not natural), see the example in Fig. 1. The example includes traffic flow and pollutants, and also some anomalies due to the dysfunction of some sensor and connections.

Please note that those data have a range of different sample rate that goes from 1 sample per minute to 1 sample for 20 minutes, not regular.

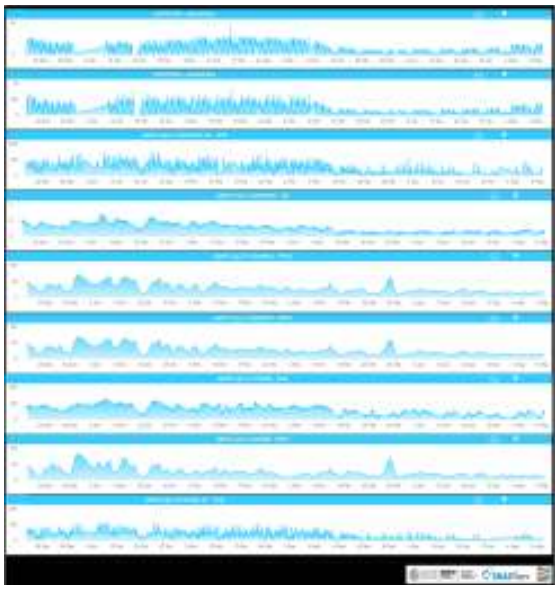

Fig. 1. Example of signals on a window from November to May 2020. https://www.snap4city.org/dashboardSmartCity/view/index.php?iddasboard=MjY5OQ==

TABLE I. Examples of monitored sensors

Metric	Category	Unit	Description
$PM_{2.5}$	Aerosol Physics	ppm	Particulate matter
PM_{10}	Aerosol Physics	ppm	Particulate matter
NO	Gaseous Pollutants	$\mu g/m^3$	Nitrogen Oxide
NO_2	Gaseous Pollutants	$\mu g/m^3$	Nitrogen Dioxide
C_6H_6	Gaseous Pollutants	$\mu g/m^3$	Benzene
SO_2	Gaseous Pollutants	$\mu g/m^3$	Sulfur Dioxide
CO	Gaseous Pollutants	ppm	Carbon Monoxide
CO_2	Gaseous Pollutants	ppm	Carbon Dioxide
O_3	Gaseous Pollutants	ppb	Ozone
H_2S	Gaseous Pollutants	$\mu g/m^3$	Hydrogen Sulfide
Temp-erature	Meteorology	°C	Air Temperature
Humidity	Meteorology	-	Air Humidity %
Average speed	Traffic flow	km/h	Avg vehicle speed
Vehicle flow	Traffic flow	vehicle/h	Vehicle flow
Concen-tration	Traffic flow	vehicle/m	Vehicles per meter

V. Data Set Modelling

On the basis of the requirements, and of the state of the art, we decided to start creating an anomaly detection algorithm for the context of smart city. This context includes cases in which data can be periodic or sporadic and are not regularly sampled, neither the periodic. The first aim has been to detect problems in the infrastructure at level of sensors, devices, storage, connections, etc., and finally also on the physical world. For the model construction, we collected 23516 data samples, each sample consisting of 20 features (10 values at consecutive timestamps, 9 time intervals of consecutive timestamps, 1 categorical feature, i.e. the sensor ID). **Table II** reports the dataset schema, where [ts1, …, ts10] are the timestamps of [value1, …, value10] respectively. We considered some air pollution and traffic related sensors (i.e., NO2, O3, CO2, PM2.5, vehicle flow).

Since we used a supervised machine learning approach, each sample has been manually labelled as normal (0) or anomalous (1). For this purpose, we used a web tool, specifically developed to label time series for each device's sensor, by just clicking on the timeframes considered anomalous. We applied a gradient boosting technique using the CatBoost algorithm [26], [27], and we split the dataset in training (2/3) and validation (1/3) sets. Since CatBoost works with categorical features out-of-the-box it is not necessary to perform a one-hot-encoding of the categorical feature. This process resulted in an unbalanced dataset with a normal/anomalous ratio of 20036/3480 (i.e., 17.36% of anomalous samples).

TABLE II. Features for learning sets

Feature	Type	Description
metric	categorical	Sensor ID
value1	numerical	value Sensor ID at t
value2	numerical	value Sensor ID at t-1
value3	numerical	value Sensor ID at t-2
value4	numerical	value Sensor ID at t-3
value5	numerical	value Sensor ID at t-4
value6	numerical	value Sensor ID at t-5
value7	numerical	value Sensor ID at t-6
value8	numerical	value Sensor ID at t-7
value9	numerical	Value Sensor ID at t-8
value10	numerical	value Sensor ID at t-9
Δt1	numerical	ts1-ts2 in ms
Δt2	numerical	ts2-ts3 in ms
Δt3	numerical	ts3-ts4 in ms
Δt4	numerical	ts4-ts5 in ms
Δt5	numerical	ts5-ts6 in ms
Δt6	numerical	ts6-ts7 in ms
Δt7	numerical	ts7-ts8 in ms
Δt8	numerical	ts8-ts9 in ms
Δt9	numerical	ts9-ts10 in ms

The approach was iterated a number of times in order to identity the satisfactory number of values over time that could be the right compromise between addressing:

- longer time series would lead to address the problems related periodicity. To that purpose the number of samples would be very high since some of them have 1 sample per minute and day and week periodicity. This means that is not affordable to take into account seasonality without resampling and creating for each data typical trends as performed in some cases at the state of the art that have been demonstrated to the insensitive to the anomalies.
- shorter time series would miss the context of the trend. 10 samples are enough to understand the last evolution less are typically not enough to detect the sample rate with the needed precision.

VI. Experimental Results

The training was performed on GPU (Nvidia Titan XP) for 10,000 iterations, using cross validation (with a validation/train split ratio of 0.33), Logloss as the loss function, accuracy as the evaluation metric, a learning rate of 0.073 and a decision tree's depth of 3. The learning rate was set to 0.0389, and the class weights applied during training were 1 and 5.673, for the normal class and the anomalous class respectively. After training, the model was shrunk to the best iteration (9835), consisting of 9836 trees. As expected, the metric's name scored among the most important features, together the time intervals between metric's values. **Table III** reports the features importance of the model, the most important features being the metric's name (i.e., the categorical feature) and the time deltas (i.e., the differences between consecutive timestamps Δt).

TABLE III. Features importance

Feature	Value
ID	3.188
Δt1	5.057
Δt2	4.690
Δt3	4.147
Δt4	3.875
Δt5	3.753
Δt6	4.232
Δt7	5.192
Δt8	3.832
Δt9	6.702
v1	1.457
v2	0.886
v3	2.015
v4	2.359
v5	1.944
v6	0.944
v7	0.818
v8	1.163
v9	1.605
v10	1.018

In addition to the above model, we used two labelling rules as representative of the rule-based solution listed above, to compare the effectiveness of the model. The first rule assumes a sequence as anomalous if data is missing for more than 1 day, i.e. a sequence is considered anomalous if Tnow - Tlast > 1 day where Tnow and Tlast are respectively the timestamps at the current time and for the last sample arrival. The second rule assumes a sequence is anomalous if data is missing for more than the median arrival time for that sensor and metric, i.e. a sequence is considered anomalous if Tnow - Tlast > Tmedian where Tnow and Tlast are respectively the timestamps at the current time and for the last sample arrival, and Tmedian is the median arrival time of data for that sensor.

Evaluation metrics reported in **Table IV** have been calculated for each model's predicted labels (i.e., the trained model and the two annotation rules), with respect to the manually annotated labels. The ML model reported the best results, in terms of accuracy, precision and recall, with respect to the labelling rules. Balanced accuracy score reports the average of recall obtained on each class (anomalous or not). The average precision reports the precision-recall curve as the weighted mean of precisions at each threshold, while the Brier Score reports the mean squared difference between the predicted probability of the possible outcomes for an item, and the real outcome. **Table IV** also reports F1 score, with macro, micro and weighted variants, being the weighted average of precision and recall. The ML algorithm provided good results,

with respect to the above-mentioned requirements, and can be deployed easily with minimal hardware requirements.

TABLE IV. EVALUATION METRICS

Model	ML	Rule # 1	Rule # 2
Accuracy	0.969	0.852	0.852
Balanced Accuracy Score	0.949	0.501	0.500
Average Precision Score	0.815	0.150	0.147
Brier Score Loss	0.030	0.147	0.147
F1 Score	[0.981, 0.896]	[0.920, 0.0057]	[0.920, 0.001]
F1 Score Macro	0.939	0.463	0.460
F1 Score Micro	0.969	0.852	0.852
F1 Score Weighted	0.969	0.784	0.783
Neg Log Loss	1.063	5.096	5.111
Precision	0.871	1.0	0.0
Recall	0.9225	0.0028	0.0
Jaccard	0.811	0.0028	0.0
ROC AUC	0.949	0.501	0.5
Vehicle flow	Traffic flow	vehicle/h	Vehicle flow
Concen-tration	Traffic flow	vehicle/m	Vehicles per meter

VII. CONCLUSIONS

This paper describes an automatic anomaly detection system for IoT solutions. The proposed solution meets the requirements described above, and is able to generalize, being able to quickly detect different types of anomalies (e.g., point, contextual, collective) related to different types of signals, even never observed, coming from different contexts (e.g., mobile, vehicles, air quality sensors), different periods and rates. It can be applied to sensors using different protocols, formats, units of measurement and types. It is a robust solution that provides a good degree of accuracy even in contexts where the measured signals are affected by noise, and is able to capture the behaviour of IoT sensors even in typical cases where they show a marked character of aperiodicity. In this sense it proves to be a valuable aid for the timely detection of faults in each level of the IoT stack (i.e., at device or individual sensor level), and in different contexts (e.g., time of day, weekday, city). In this regard, it should be pointed out that, in a rapidly evolving IoT context, with an increasing number of sensors of different types and reliability, installed in different smart city contexts, such an approach may require periodic training, in order to ensure that the model is updated to the new dynamics of the observed signals.

ACKNOWLEDGMENT

The authors want to thank all the peers involved in the Select4Cities project (https://www.select4cities.eu) and the European Commission. European Commission Affiliation Select4Cities project has received funding from the European Union's Horizon 2020 research and innovation programme under the grant agreement N. 688196.

REFERENCES

[1] V. Chandola, A. Banerjee, V. Kumar, "Anomaly detection: A survey", ACM Comput. Surv., vol. 41, no. 3, pp. 15, 2009.

[2] Z. Niu, S. Shi, J. Sun, X. He, "A survey of outlier detection methodologies and their applications", Proc. Int. Conf. Artif. Intell. Comput. Intell, pp. 380-387, 2011.

[3] Xiao-yun Chen, Yan-yan Zhan, Multi-scale anomaly detection algorithm based on infrequent pattern of time series, Journal of Computational and Applied Mathematics, Elsevier, Volume 214, Issue 1, 15 April 2008, p. 227-237

[4] X. Song, M. Wu, C. Jermaine, S. Ranka, "Conditional anomaly detection", IEEE Trans. Knowl. Data Eng., vol. 19, no. 5, pp. 631-645, May 2007.

[5] S. M. Mahmoud, A. Lotfi, C. Langensiepen, "Abnormal behaviours identification for an elder's life activities using dissimilarity measurements", Proc. 4th Int. Conf. Pervasive Technol. Rel. Assistive Environ., pp. 25, 2011.

[6] P. Cuddihy, J. Weisenberg, C. Graichen, M. Ganesh, "Algorithm to automatically detect abnormally long periods of inactivity in a home", Proc. 1st ACM SIGMOBILE Int. Workshop Syst. Netw. Support Healthcare Assist. Living Environ., pp. 89-94, 2007.

[7] F. Cardinaux, S. Brownsell, M. Hawley, D. Bradley, "Modelling of behavioural patterns for abnormality detection in the context of lifestyle reassurance", Proc. Iberoamerican Congr. Pattern Recognit, pp. 243-251, 2008.

[8] W. Kang, D. Shin, D. Shin, "Detecting and predicting of abnormal behavior using hierarchical Markov model in smart home network", Proc. IEEE 17Th Int. Conf. Ind. Eng. Eng. Manage. (IE&EM), pp. 410-414, Oct. 2010.

[9] F. J. Ordóñez, P. de Toledo, A. Sanchis, "Sensor-based Bayesian detection of anomalous living patterns in a home setting", Pers. Ubiquitous Comput., vol. 19, no. 2, pp. 259-270, 2015.

[10] V. R. Jakkula, D. J. Cook, "Detecting anomalous sensor events in smart home data for enhancing the living experience", Proc. Artif. Intell. Smarter Living, pp. 1-2, 2011.

[11] A. Palaniappan, R. Bhargavi, V. Vaidehi, "Abnormal human activity recognition using SVM based approach", Proc. Int. Conf. Recent Trends Inf. Technol. (ICRTIT), pp. 97-102, Apr. 2012.

[12] J. H. Shin, B. Lee, K. S. Park, "Detection of abnormal living patterns for elderly living alone using support vector data description", IEEE Trans. Inf. Technol. Biomed., vol. 15, no. 3, pp. 438-448, May 2011.

[13] N. Han, S. Gao, J. Li, X. Zhang, J. Guo, "Anomaly detection in health data based on deep learning", Proc. Int. Conf. Netw. Infrastruct. Digit. Content (IC-NIDC), pp. 188-192, Aug. 2018.

[14] L.-J. Chen, Y.-H. Ho, H.-H. Hsieh, S.-T. Huang, H.-C. Lee, S. Mahajan, "ADF: An anomaly detection framework for large-scale PM2.5 sensing systems", IEEE Internet Things J., vol. 5, no. 2, pp. 559-570, Apr. 2018

[15] W. Yan, "One-class extreme learning machines for gas turbine combustor anomaly detection", Proc. Int. Joint Conf. Neural Netw. (IJCNN), pp. 2909-2914, Jul. 2016.

[16] D. Kumar, J. C. Bezdek, S. Rajasegarar, M. Palaniswami, C. Leckie, J. Chan, J. Gubbi, "Adaptive cluster tendency visualization and anomaly detection for streaming data", ACM Trans. Knowl. Discovery Data (TKDD), vol. 11, no. 2, pp. 24, 2016.

[17] M. A. Hayes, M. A. Capretz, "Contextual anomaly detection in big sensor data", Proc. IEEE Int. Congr. Big Data (BigData Congr.), pp. 64-71, Jun. 2014.

[18] Randeep Bhatia, Steven Benno, Jairo Esteban, T. V. Lakshman, and John Grogan. Unsupervised machine learning for network-centric anomaly detection in IoT. In Proceedings of the 3rd ACM CoNEXT Workshop on Big DAta, Machine

Learning and Artificial Intelligence for Data Communication Networks (Big-DAMA '19).
[19] Y. Cheng, Y. Xu, H. Zhong and Y. Liu, "HS-TCN: A Semi-supervised Hierarchical Stacking Temporal Convolutional Network for Anomaly Detection in IoT," 2019 IEEE 38th Int. Performance Computing and Communications Conference (IPCCC), London, United Kingdom, 2019, pp. 1-7.
[20] Yong Yu, Xiaosheng Si, Changhua Hu and Jianxun Zhang, A Review of Recurrent Neural Networks: LSTM Cells and Network Architectures, Neural Computation Volume 31, Issue 7, July 2019 p.1235-1270
[21] Sagheer, Kotb 2019] Sagheer, A., Kotb, M. Unsupervised Pre-training of a Deep LSTM-based Stacked Autoencoder for Multivariate Time Series Forecasting Problems. Sci Rep 9, 19038 (2019).
[22] T. V. Duong, H. H. Bui, D. Q. Phung, S. Venkatesh, "Activity recognition and abnormality detection with the switching hidden semi-Markov model", Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR), vol. 1, pp. 838-845, Jun. 2005.
[23] C. Badii, et al., "Snap4City: A Scalable IOT/IOE Platform for Developing Smart City Applications", Int. Conf. IEEE Smart City Innovation, China 2018, IEEE Press. DOI: https://ieeexplore.ieee.org/document/8560331/
[24] C. Badii, P. Bellini, A. Difino, P. Nesi, "Smart City IoT Platform Respecting GDPR Privacy and Security Aspects." IEEE Access 8 (2020): 23601-23623.
[25] P. Bellini, S. Bilotta, P. Nesi, M. Paolucci, M. Soderi, "Real-Time Traffic Estimation of Unmonitored Roads", IEEE-DataCom'2018, Athen, 2018.
[26] L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, A. Gulin, CatBoost: unbiased boosting with categorical features, Thirty-second Conference on Neural Information Processing Systems, NeurIPS 2018.
[27] A. V. Dorogush, V. Ershov, A. Gulin, CatBoost: gradient boosting with categorical features support, Workshop on ML Systems at NIPS 2017.