Anomaly Detection on IoT Data for Smart City

Pierfrancesco Bellini, Daniele Cenni, Paolo Nesi

{pierfrancesco.bellini, daniele.cenni, paolo.nesi}@unifi.it

Distributed Systems and Internet Technology Lab, http://www.disit.dinfo.unifi.it Department of Information Engineering, http://www.dinfo.unifi.it University of Florence, http://www.unifi.it, Florence, Italy, phone +39-3355668674

Email: {pierfrancesco.bellini, daniele.cenni, paolo.nesi}@unifi.it

SNAP4CITY KM4CITY

Context and motivation

- Research conducted in the context of Smart City management
- Goal: enhance the reliability of IoT infrastructures
- Develop tools and methods to detect, collect and classify IoT devices related anomalies
- We are interested in estimating how IoT devices behave, we want to detect anomalous IoT devices *patterns* and outliers, studying aggregated data
- The aim is to collect data to be able to build a model that can analyze IoT devices temporal sequences to identify anomalous data
- Applications: control and regulation of IoT devices, detection of bottlenecks, predictions on harmful behaviors, optimization of IoT infrastructures

Context and motivation

- Anomaly detection analysis is a major prerequisite for planning IoT devices maintenance
- European Commission on IoT: "*data should be made available, accessible and easily aggregated, processed, as well as trusted, thus combining quality, **reliability** and security*" ("Advancing the Internet of Things in Europe").

Source: https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:52016SC0110

- Understand how IoT devices behave and exploit IoT services with different AI techniques is a major topic
- Develop a methodology for the detection of IoT sensors *outliers*, to effectively identify and model different types of anomalies

• This research was conducted in the context of the Snap4City PCP Select4Cities projects (https://www.snap4city.org, https://www.select4cities.eu)

IoT Anomalies

- *Point anomaly*, is nothing more than an outlier, i.e. a value significantly different in size than the rest of the data;
- *Contextual anomaly* (also called conditional anomaly), 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.

Techniques

- How to detect anomalies and outliers?

• Many possible solutions: dissimilarity measures, Gaussian Mixture models, Hierarchical Markov models, Bayesian models, Switching Hidden Semi-Markov models, Support Vector Machines, Support Vectors, Convolutional Neural Networks and Recurrent Neural Networks, Extreme Learning Machines, Clustering, Multivariate Clustering, semi-supervised hierarchical stacking Temporal Convolutional Network (TCN), LSTM networks and LSTM Autoencoders to reconstruct time series and find out if they resemble a normal behaviour

- Most of the most up-to-date approaches use machine learning techniques that require to collect a large amount of data to be able to build a model that can effectively generalize from data
- Data must be obtained by collecting time series from each IoT sensor in the IoT infrastructure

Preparing Data for Training

This is the most challenging task

- Data must be crawled for a sufficient period of time;
- Data must be collected from a heterogeneous set of IoT sensors, in order to be able to generalize when using the model with new data from unseen IoT sensors;
- Data must be normalized (we are dealing with many different sensors and units of measure);
- IoT data are not periodical, since IoT sensors update the measurement only if it changes with respect to the previous value;
- IoT data do not show a trend or seasonality, neither in a long timeframe: clustering techniques do not offer the same degree of accuracy as manually labelling techniques;
- Dataset cleanup: for the construction of the dataset, IoT sensors must be carefully evaluated and included only if they consistently provide reliable data. Sensors that are always faulty must be avoided in order not to dirty the dataset;
- At the end of the annotation task we get an unbalanced dataset, calculating the weighting factor is fundamental for the construction of a good model.

IoT Data

- Data are not periodically sampled. For economy reasons IoT sensors are programmed in a way such to avoid sending messages when the data value has not changed significantly

Requirements Analysis

An ideal solution for anomaly detection should be capable to detect sensor related anomalies even if they:

- (*noise*) are 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;
- are confined in one point (*outlier*) with respect to the typical bounds;
- are 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 the context is also needed;
- (*typical trends*) are referring to some classification and typical trends that are not met. This implies to take into account the context and seasonal trends;

Requirements Analysis

An ideal solution for anomaly detection should be capable to detect sensor related anomalies even if they are:

- (*period*) signals that could be taken periodically or sporadically. In the context of Smart City and IoT 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;
- (*rate*) signals that could present different sampling rates. 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;
- (*scalable*) the solution has to be scalable to avoid investing huge amount of resources in anomaly detection in real time.

Requirements Analysis

An ideal solution for anomaly detection should be capable to detect sensor related anomalies even if they:

- (*structure*) belong to IoT devices with multiple sensors, and the single IoT devices may belong to collection of IoT devices. In 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;
- (*moving*) are located on IoT devices that are moving, such as Mobile App, vehicles, air quality sensors located on busses, etc.;
- (*producer*) belong to IoT sensors produced by different builders/producers, protocols, data formats, unit of measures, data types, sample rates, etc.;
- (*stack faults*) are due to different causes/faults along the IoT stack.

Data Analysis

• In the context of the present work, data was crawled from air quality and traffic related devices (air quality pollutants, weather conditions, and traffic flow data). For example:

Metric	Category	Unit	Type	Description
$PM_{2.5}$	Aerosol Physics	ppm	float	Particulate matter (2.5µm)
PM_{10}	Aerosol Physics	ppm	float	Particulate matter (10µm)
NO	Gaseous Pollutants	$\mu g/m^3$	float	Nitrogen Oxide
NO_2	Gaseous Pollutants	$\mu g/m^3$	float	Nitrogen Dioxide
C_6H_6	Gaseous Pollutants	$\mu g/m^3$	float	Benzene
SO_2	Gaseous Pollutants	$\mu g/m^3$	float	SulfurDioxide
CO	Gaseous Pollutants	ppm	float	Carbon Monoxide
Concentration	Traffic flow	vehicle/m	integer	Vehicles per m

Data Analysis

- In the context of the present work, data was crawled from air quality and traffic related devices (air quality pollutant, weather conditions, and traffic flow data):

Note: data have different sample rates that go from 1 sample per minute to 1 sample per 20 minutes

Metric	Category	Unit	Type	Description
CO_2	Gaseous Pollutants	ppm	float	Carbon Dioxide
O_3	Gaseous Pollutants	ppb	float	Ozone
H_2S	Gaseous Pollutants	$\mu g/m^3$	float	Hydrogen Sulfide
Temperature	Meteorology	°C	float	Air Temperature
Humidity	Meteorology	-	float	Air Humidity (%)
Average speed	Traffic flow	km/h	float	Average vehicle speed
Vehicle flow	Traffic flow	vehicle/ h	float	Vehicle flow

Data Set Modelling

- We decided to start creating an anomaly detection model for the context of smart city
- 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)

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

Data Set Modelling

- We decided to start creating an anomaly detection model for the context of smart city
- 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)

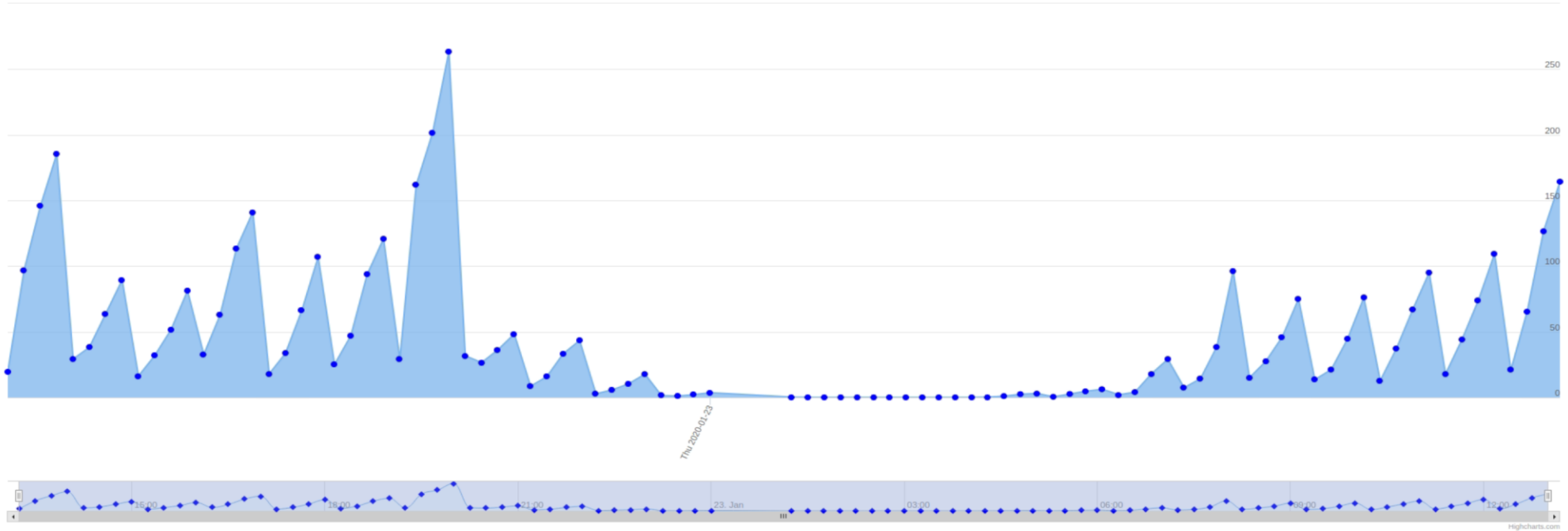
Feature	Type	Description
Δt1	categorical	ts1-ts2 in ms
Δt2	numerical	ts2-ts3 in ms
Δt3	numerical	ts3-ts4 in ms
Δt4	numerical	ts4-ts5 in ms
Δt5	numerical	ts5-ts6 in ms
Δt6	numerical	ts6-ts7 in ms
Δt7	numerical	ts7-ts8 in ms
Δt8	numerical	ts8-ts9 in ms
Δt9	numerical	ts9-ts10 in ms

Model building

- 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. The dataset was split 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).

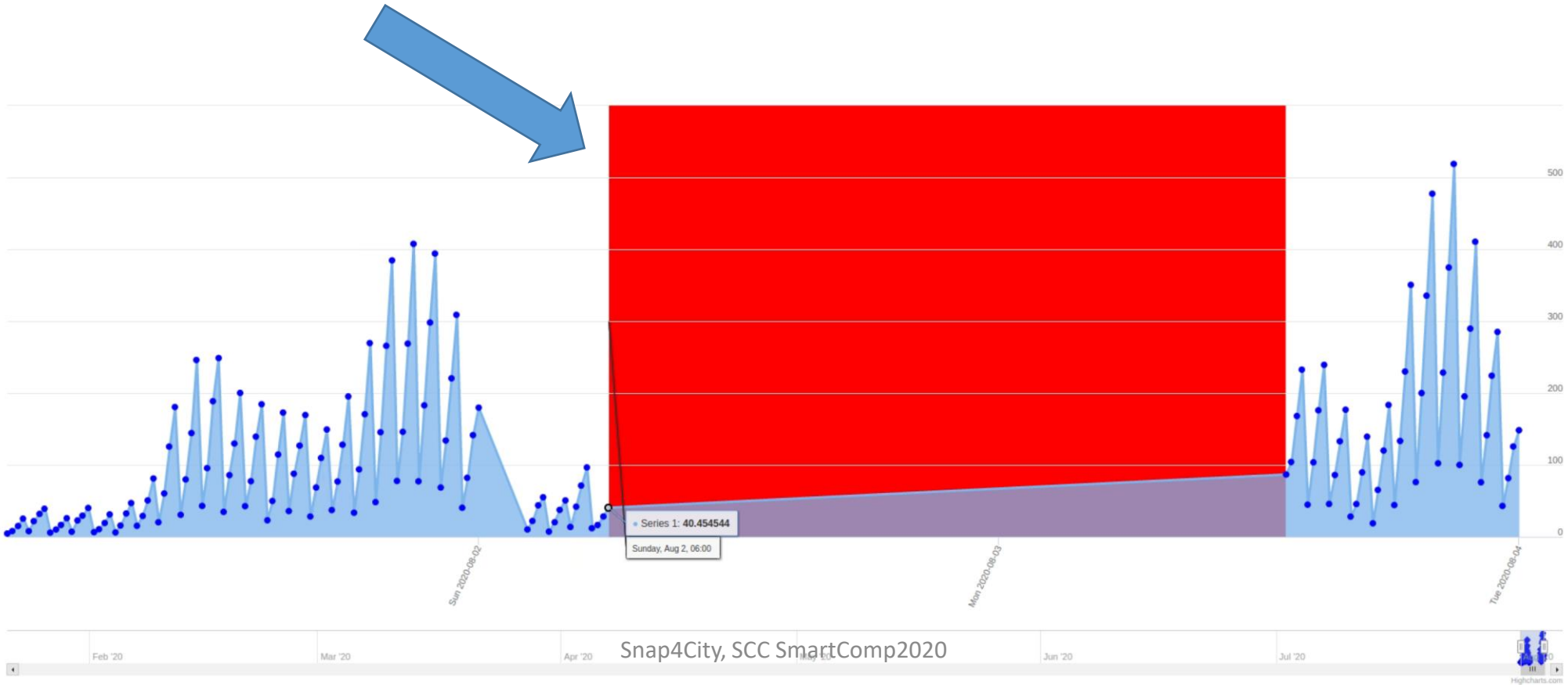
Human Data Annotation

- The user can select IoT sequences by date range and manually annotate patterns considered anomalous or containing outliers



Human Data Annotation

- In this case there is lack of data for a considerable amount of time: marked as anomalous



Experimental Results

- 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.

Data issues

The approach was iterated a number of times in order to identify 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 to periodicity. To that purpose the number of samples would be very high since some of them have 1 sample per minute and day or week periodicity. This means that is not affordable to take into account seasonality without resampling and creating for each data typical trends;
- *shorter time series* would miss the context of the trend, 10 samples are enough to understand the last evolution, while less are typically not enough to detect the sample rate with the needed precision.

Experimental Results

Features importance

- We measure the relative importance of each feature when making a prediction
- the most important

features are the metric's name (i.e., the categorical feature) and the time deltas (i.e., the differences between consecutive timestamps Δt)

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

Feature	Value
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

Other Experiments

- What about if we measure the accuracy with a machine learning approach with that of a traditional one?
- 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;
- *First rule*, assumes a sequence as anomalous if data is missing for more than 1 day, i.e. a sequence is considered anomalous if $T_{now} - T_{last} > 1$ day where T_{now} and T_{last} are respectively the timestamps at the current time and for the last sample arrival;
- *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 $T_{now} - T_{last} > T$ median where T_{now} and T_{last} are respectively the timestamps at the current time and for the last sample arrival, and T median is the median arrival time of data for that sensor.

Experimental Results

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

Model	ML	Rule #1	Rule #2
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

- Evaluation metrics 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;

- We calculated the balanced accuracy score, reporting the average of recall obtained on each class (anomalous or not), the average precision, reporting the precision-recall curve as the weighted mean of precisions at each threshold, the Brier Score reporting the mean squared difference between the predicted probability of the possible outcomes for an item, and the real outcome, the F1 score, with macro, micro and weighted variants, being the weighted average of precision and recall;
- The ML model provided good results, with respect to the above-mentioned requirements, and can be deployed easily with minimal hardware requirements.

Thank you!