Consensus-based Anomaly Detection for Efficient Heating Management

Francesco Carbone
Dept. of Information Engineering
University of Padova
35131 Padova, Italy
Email: francesco.carbone.1@unipd.it

Angelo Cenedese
Dept. of Information Engineering
University of Padova
35131 Padova, Italy
Email: angelo.cenedese@unipd.it

Cinzia Pizzi
Dept. of Information Engineering
University of Padova
35131 Padova, Italy
Email: cinzia.pizzi@unipd.it

***Abstract*—The analysis of data to monitor human-related activities plays a crucial role in the development of *smart* policies to improve well being and sustainability of our cities. For several applications in this context anomalies in time series can be associated to smaller timeframes such as days or weeks.**

In this work we propose a consensus-based anomaly detection approach that exploits the power of the Symbolic Aggregate approXimation (SAX) and the specificity of such time series. In our approach, the normalization of the signal becomes a proper element of the modeling. In fact, we conjecture that different normalization horizons allow to include in the shape of the timeseries patterns an additional, variable, component from a longer period trend. To support the analysis phase, a calendar can be used as an additional source of information to discriminate between really unwanted anomalies and expected anomalies (e.g. weekends), or even to signal a possible anomaly whenever a "normal" behavior is not expected.

Preliminary experiments on temperature analysis in an indoor environment, with the scope of thermal energy saving, showed that our approch effectively identifies all known anomalies, and also pointed out some unexpected, but clear, anomalies.

***Keywords*-Time series; Anomaly detection; Data mining; Sequence analysis;**

I. Introduction and related work

Nowadays, pervasive networking of sensors and actuators has definitely changed our way of interacting with the environment, thanks to the advances in technology and novel paradigms in distributed system theory as well as in information and coding theory. Indeed, these devices can offer access to an unprecedented quality and quantity of information that can revolutionize our ability in controlling the human space (see for example the survey [1]). This revolution has benefited from the synergy among several disciplines, and the applications that derive cover a wide variety of fields and objectives, from enhancing productivity in an industrial framework, to assisting everyday life, to supporting operators in critical scenarios, just to cite a few [2].

In this context, one of the most promising and fascinating field of research regards the *smart city* framework where the acts of circulating multilevel heterogeneous information through network systems and of understanding its content are of key importance to build a livable environment. Particularly, the advances in analysis of data to monitor human-related activities play a crucial role in the development of *smart* policies to improve well being and sustainability of our cities. Given the huge volume of data that are available to the user, the automatic interpretation of time series in search for anomalies or with the aim of building a normality space are in order, motivating an ever increasing research by both the academic and the industrial communities on time series data mining [3] and anomaly detection [4].

Given these premises, our work is motivated by the problem of anomaly detection in temperature time series obtained from a sensor network deployed in a primary school. The aim of the application is to detect possible waste of energy, which is a canonical problem in the context of smart building policies. Remarkably, such kind of time series is strictly related to human activity as the main factors that affect the indoor measured temperature are the human presence and the daily human activity on the one side, and the seasonal climate change on the other. In such a context, we have an additional source of information that allow to detect really unwanted anomalies and drop anomalies that are indeed expected (weekends, festivities, etc). Although the application that motivated us considers real-value time series, the approach proposed in this work is based on a symbolic representation of the time series, and on an iterative procedure where a random sampling allows to iteratively build a model and operate the anomaly detection. Then, a voting procedure determines the actual model and retrieved anomalies, in this taking inspiration from consensus problems that arise in large scale problems with noisy inputs [5].

The choice of using a symbolic representation of the input, so that anomalies will be actually extracted from a string of symbols defined over a given finite alphabet, is driven by the reduction of both noise and data dimensionality that it provides. Among the possible techniques for sequence

Presented at IEEE SCI 2017
International Workshop on Big Data Applications for Smart Cities (BDASC 2017)

discretization, we chose to use the Symbolic Aggregate approXimation (SAX) [6], which proved to be successfull in a large number of applications [7].

In the context of anomaly detection on discrete sequences there are three main problem formulations [8]:

1) Sequence-based: find the most anomalous sequence(s) among a given set of sequences;
2) Subsequence-based: given a sequence s, find the most anomalous subsequence of length w with respect to the other subsequences of s;
3) Pattern frequency-based anomaly detection: a test pattern is anomalous if the difference between its frequency in a test sequence s and its frequency in a set of training sequences is above a given threshold.

For each of these formulations, there is a large number of approaches that can be used to solve the problem of anomaly detection (e.g. [9], [10], [11]). We refer to [8], [12] for an indepth overview of the state-of-the art on time series characterization.

Our problem is at the intersection between the sequence and the subsequence based formulations. While we are interested in the detection of anomalous frames of a specific length (e.g. a day or a week), we are not interested in the analysis of all the subsequences, but rather on the subset of adjacent subsequences of that length.

The approach we are going to follow to solve this problem aims at the combination of powerful tools, such as the aforementioned SAX representation, and the consensus clustering approach [13], [14], in that the procedure of model building searches for a consensus among the data by "guessing and voting" the model and the anomalies proposed by several independent k-means clusterings. Note that, since we cluster *non-overlapping* subsequences, we do not risk to fall into meaningless clustering [15].

A peculiarity of our approach is the decoupling of the normalization and discretization phases. In fact, it is almost dogmatic in anomaly detection that the sequences to be compared are normalized with respect to their own mean and standard deviation. While this is justified when shape is the only feature that matters, there might as well be applications (such as the one we studied) in which the absolute value of the data is also significant.

To support the analysis of human-related activities, we integrated the use of a calendar that will allow us to better discriminate the kind of anomalies that are reported. The integration of supplementary information in the process of anomaly detection has been formalized in a seminal paper [16] defining the characteristics of *conditional anomaly detection*. However, our approach presents several differences with respect to that definition. In [16] a generative model is build assuming that baseline data are available, and that attributes can be partitioned in two classes: *indicator* attributes (those that actually reveal the anomaly), and environmental attributes (the additional information that might justify the presence of an anomaly). Both kind of attributes are used to build the model, and the role of environmental attributes is mainly that of reducing the number of false positives. On the contrary, ours is a discriminative approach, based on sequence similarity, in which the additional information (the calendar) is not used to build the model, but as an additional post-processing "check". Moreover, although clearly specific for this kind of data analysis, the calendar helps to signal the presence of both false positives and false negatives (e.g. normal behaviors during festivities), the latter being probably the hardest kind of anomaly to detect.

The rest of the paper is organized as follows. In Sec. II the main contribution of this work is presented and discussed. The experimental results are reported in Sec. III and finally, conclusions and future work are discussed in Sec. IV.

II. Consensus-based Anomaly Detection

The approach proposed here considers the SAX approximation of a time series and is based on an iterative procedure to cluster frames and detect the anomalies. A schematic pipeline of this approach is shown in Fig.1, where three main phases are emphasized, and separately discussed below.

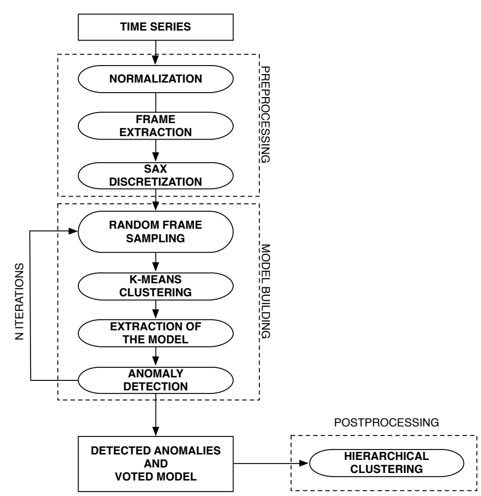

Figure 1. Consensus-based anomaly detection processing steps.

A. Signal Preprocessing

The signal in our specific case of indoor monitoring is the output of an analog sensor sampling the room temperature every 5 minutes. To translate it into a symbolic sequence, we employ the SAX representation [6], which is a widely used solid and flexible method to obtain a string-based representation of a time series. In particular, SAX allows to reduce the signal dimensionality and guarantees a lower bound of the L_p norm. This is an important feature as it makes possible to apply data mining algorithms to the time series represented through SAX, producing the same results as those that the algorithms would do if fed with the original time series.

The translation of the real values of the data stream into a set of symbolic string is done in three steps: normalization, frame extraction, and SAX discretization.

Normalization: Generally speaking, normalization is necessary to allow for comparison of different time series [17]. However, applying normalization to a set of time series has the effect of making their shapes comparable, while for some applications the difference in terms of absolute value might be significant as well. This situation easily occurs in the environmental monitoring data where, for example, in a temperature time series two different days might have the same temperature trend, but different values. With respect to the scenario of our application, this could point out that on both days the heating system is on (justifying the similar trend), but only in one of them there is human presence that increases the temperature values through passive contribution but at the same time motivates the fact of having the heating on.

The normalization process takes an input signal $T = t_1, \ldots, t_n$ and outputs a signal $\widetilde{T} = \tilde{t}_1, \ldots, \tilde{t}_n$, where:

$$\tilde{t}_i = \frac{t_i - \mu}{\sigma} \quad , \qquad i = 1, ..., n, \tag{1}$$

being μ and σ the average and standard deviation of the signal over a chosen horizon.

Usually, when comparing subsequences taken from a time series, these two values μ and σ are computed on each subsequence, per-segment, just before the discretization step: this allows to compare time series solely in terms of shape. However, when dealing with subsequences that are in turn "building blocks" of longer and meaningful subsequences, it may be beneficial to perform the computation of mean and standard deviation over such longer subsequences, to highlight more complex patterns. Thus, we conjecture that normalization may be applied differently from the standard approach, and the choice of the normalization horizon is induced by the possibility of detecting characteristic periods along the data streams (e.g. days, weeks, months) and by the type of analysis we want to perform on the data. In this sense, we discuss here the case of interest for the environmental analysis, such the one motivating this work, and in which the time series can be partitioned according to days, weeks, and months:

- *Whole Series Normalization*: the input time series T is used to compute the average μ and the standard deviation σ; Each t_i is then normalized according to (1).
- *Monthly Normalization*: the input time series T is partitioned into months. The average and standard deviation are compututed with respect to each month, which is then normalized separately. This means that a t_i belonging to a specific month is normalized with respect to its corresponding month average and standard deviation, so as to maintain its seasonal characteristics;
- *Weekly Normalization*: the time series T is partitioned into weeks, and each week is normalized separately, according to its specific average and standard deviation. This normalization is useful to highlight the presence in a "normal" week of both weekdays and weekends, which usually show strongly different trends;
- *Daily Normalization*: The time series T is partitioned into days, and each day is normalized separately, according to its specific average and standard deviation. Note that, since the day coincides with the segment of interest in our analysis, this strategy is equivalent to the one traditionally applied when discretizing with SAX, as the normalization (1) is specific of each extracted frame.

In Fig. 2, it can be appreciated the result of different normalizations of the same time series: with the Daily Normalization, all days show the same pattern due to daytime-nightime alternation, while in the Weekly Normalization, the information about weekdays and weekends is preserved as in the original series, gaining the attenuation of the seasonal trend.

Frame Extraction: Independently on the chosen kind of normalization we can analyze frames of a different length. For example, we might want to detect anomalous days, having normalized the sequence by weeks. In this step, the normalized sequence $\widetilde{T}$ is simply partitioned into frames of length L, where L is the number of samples in our unit of observation.

SAX Discretization: Each of the resulting time frames will then be discretized with SAX. Let $\tilde{\tau} = \tilde{\tau}_1 \ldots \tilde{\tau}_n$ be one of such frames. The SAX algorithm first transforms the time series $\tilde{\tau}$ into a series $\bar{\tau} = \bar{\tau}_1, \ldots, \bar{\tau}_w$ of size w, where $w \ll n$. Essentially, the average value of the data falling within a frame is calculated and a vector of these values becomes the data-reduced representation. This procedure is called Piecewise Aggregate Approximation (PAA) [18].

The PAA representation is then replaced by a string of symbols, each of which is taken from an alphabet $\Sigma = \{\alpha_1, \ldots, \alpha_a\}$ of size a. To correctly reflect the time series characteristics, the discretization will produce equiprobable symbols by dividing the area under the normal distribution $\mathcal{N}$ into a regions bounded by breakpoints β_i, so that $\beta_{i+1} - \beta_i = \frac{1}{a}$, and coefficients of the PPA between β_i and β_{i+1} are mapped to the symbol α_{i+1}. The resulting concatenation of w symbols is called *word* and indicated as

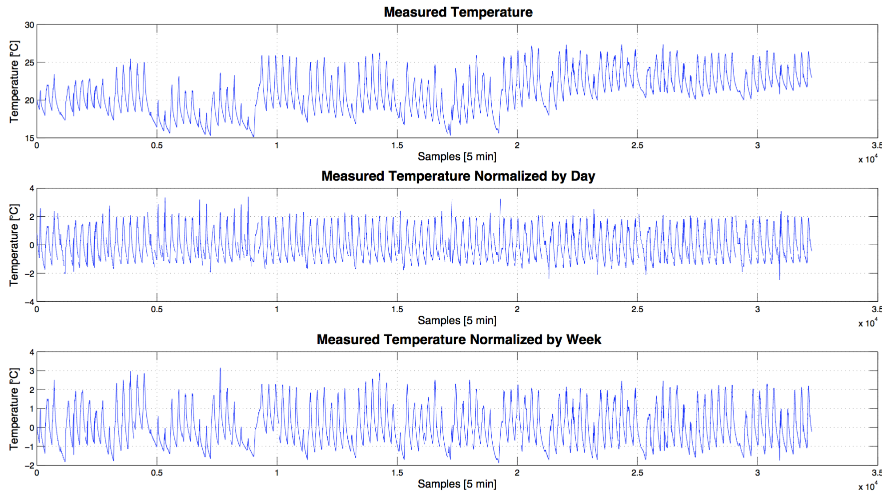

Figure 2. Top: input time series $\widetilde{T}$. Middle: the series normalized by day. Bottom: the same input series normalized by weeks.

$\widehat{T}$. Fig. 3 shows an example of the SAX representation.

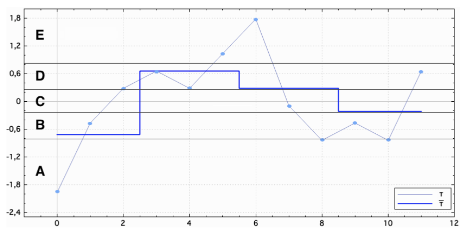

Figure 3. Example of SAX discretization on a segment of length $n = 12$. The original signal T and its PAA representation $\overline{T}$ are shown. In the example $w = 4$ and $a = 5$: the resulting word $\widehat{T}$ encoding the time series is bddc.

The distance d_{SAX} between two SAX words $\widehat{X} = x_1 \dots x_w$ and $\widehat{Y} = y_1 \dots y_w$ is: $d_{SAX}(\widehat{X}, \widehat{Y}) = \sqrt{\frac{n}{w}} \cdot \sqrt{\sum_{i=1}^{w} (\mathrm{dist}(\widehat{x}_i, \widehat{y}_i))^2}$, where $\mathrm{dist}(*, *)$ is the minimal distance between two symbols:

$$\mathrm{dist}(i, j) = \begin{cases} 0 & \text{if } |i - j| \leq 1 \\ \beta_{\max(i,j)-1} - \beta_{\min(i,j)} & \text{otherwise} \end{cases} \quad (2)$$

Remarkably, the distance between two symbols solely depends on the breakpoints, so it is possible to pre-compute the distances between simbols and store them in a look-up table.

B. Consensus-based Anomaly Detection and Model Building

After pre-processing, the model to characterize "normal frames" is extracted from the set F of symbolic strings corresponding to the frames of interest.

Basic Clustering-based Solution: Starting from the key assumption that normal data belong to large and dense clusters, while anomalies do not belong to any cluster or form very small clusters, clustering can be used to discover anomalies [4].

In its simplest implementation, we apply the k-means algorithm to the set F. Each cluster C_i within F is identified by a centroid c_i and the k-means algorithm adopts the d_{SAX} distance to compute the objective function φ to minimize and find the optimal partition $\{C_i\}^*$:

$$\{C_i\}^* = \arg\min_{\{C_i\}} \sum_{i=1}^{k} \sum_{\widehat{T} \in C_i} d_{SAX}(\widehat{T}, \widehat{c}_i). \quad (3)$$

To free the user from the selection of an additional parameter we adopted a local search strategy within a range of values of k, by adapting the method described in [19] to our framework. Clusters with a small cardinality (<20% of the size of the frameset) and far from other centroids are discarded to reduce the probabilty of having anomalies affecting the model. The final model is then composed by the strings that correspond to the centroids of the remaining (optimal) clusters: $\mathcal{M} = \{\widehat{c}_1^*, \dots, \widehat{c}_r^*\}$.

The model is then used to i) search for anomalies:

$$\widehat{T} \text{ is an anomaly} \quad iif \quad \min_{\widehat{c}_j^* \in \mathcal{M}} d_{SAX}(\widehat{T}, \widehat{c}_j^*) > Z \quad (4)$$

where Z is the given treshold, and ii) to label the environment itself for possible subsequent analysis.

Increasing Robustness: Sample and Iterate: In an unsupervised setting, the set F might contain anomalies. The procedure based on the cardinality and/or the distance among clusters can only mitigate this phenomenon. To furtherly reduce the contribution of anomalies we iterate the procedure described before on N independent samplings F_i of F. For each F_i we proceed as described above, dropping clusters with a small cardinality and far away centroids. The resulting model $\mathcal{M}_i$ contains the remaining centroids and will be used to detect anomalies that will be stored in a list L_i.

Anomaly and Model Selection through Consensus: Starting from the N model-anomalies pairs $(\mathcal{M}_i, L_i)$, a majority voting scheme is built that will determine the "best" anomalies and the "best" model, looking for a consensus among the different sets $\{\mathcal{M}_i\}$ and $\{L_i\}$.

Given a quorum $q \leq N$, a list L is obtained by selecting all the anomalous frames ($\widehat{T} \in L_i$, for some i) that have been detected by at least q models $\mathcal{M}_i$ out of N.

On the other hand, the models that contribute to the detection of the "best" (highest ranked) anomalies are reported as the "best" models for the scenario to describe the normal days, and are associated to the specific environment.

Exploiting Additional Information: In many situations, although the scenario remains not supervised, some additional information is available for the nature itself of the data. For example, in indoor enviromental monitoring, the data are strictly correlated with the human presence and activities, which are regulated by calendar times and dates.

This aspect appears remarkable since it yields the possibility of exploiting prior information about expected systematic anomalies (as weekends or holidays or closing times). During the model building phase these events can be discarded from the whole set F to avoid their contribution to build the "correct" normality model, or they can be grouped together to form an alternative normality. During the detection phase, they can be filtered out to prevent the generation of false alarms.

C. Anomaly Visualization and Postprocessing

The designed tool allows for classical anomaly visualizations, as a list, or for a synoptical view on the original signal highlighted with color code.

Moreover, it is also possible to further process the output by clustering the results with a hierarchical algorithm, facilitating the human analysis. An example is shown in Fig. 4.

III. Preliminary Experiments

In this section we present experimental results aiming at the assessment of the proposed procedure.

The time series analysis is considered with respect to different characteristic pattern lengths, namely the daily and the weekly basis, to highlight possible anomalies occurring at different timescales. It is useful to recall the aforementioned notions of *warning* and *alarm*: while normally any detected anomaly triggers an alarm, when additional external information (e.g. a calendar) is supplied, it is reasonable to assume that some anomalies can actually be accepted (e.g. they refer to festivities) and thus they are linked to a warning action.

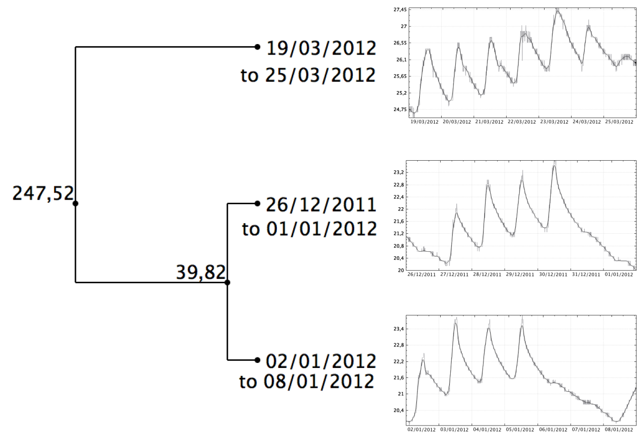

Figure 4. Hierarchical clustering of retrieved anomalies. The distance between clusters is on the edge of the dendogram. The weeks closer to each other are both holidays weeks, while the other does not correspond to any festivity.

A. Case study description

The environmental data considered in this work refer to the indoor temperature signals measured in a primary school during a period of 126 days (18 weeks) from December 8th, 2011 to March 26th, 2012. These data are sampled every 5 minutes and are obtained from a wireless sensor network (WSN) deployed in rooms of different sizes, occupancy, and use. In Fig. 5 the map of the studied environment is shown.

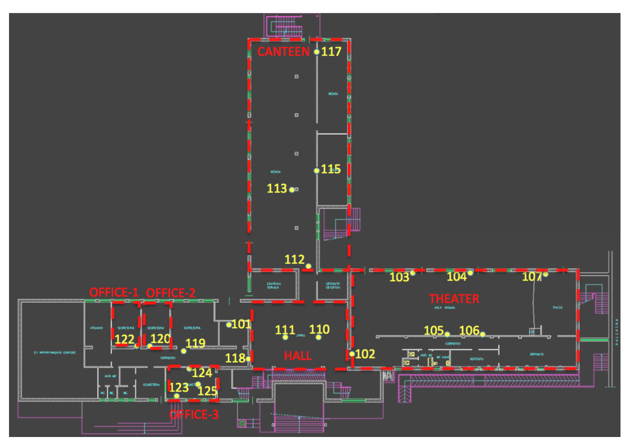

Figure 5. Case study scenario. Indoor environmental monitoring in a primary school with a WSN. The sensor nodes are shown as yellow dots, while the main rooms are highlighted and labeled with red marks.

The value of the anomaly threshold Z is a convex combination of pattern length and alphabet size, with weights respectively 0.2 and 0.8, to account for both the time discretization of the reference period and the description sensitivity. Z is set to 9.6 and 40 for the daily and weekely analysis, respectively. The sampling for the model building is repeated N=5 times, and we require the majority of votes (therefore q=3) to select a model. The other parameters are discussed in the following subsections.

B. Daily Analyses

When looking for anomalies on a daily basis, the choice of the alphabet size a is related to the detection sensitivity (the larger the size, the more sensitive the detection), but it does not affect the detection performance if it is restricted to a reasonable range $a \in [5, 16]$. This aspect can be justified by observing that the normal behaviors are quite similar in shape, therefore to detect a potential anomaly a large granularity can be tolerated. To represent the single day, the pattern length w has been chosen equal to 24, with the rationale of assigning one symbol per hour, and the normalization is performed day by day, so as to characterize the day with the shape of the temperature profile.

The results, reported in Fig. 6, show the good performance of the procedure, by revealing the days when the heating system is turned off or when the temperature profile shows an uncommon shape with respect to a normal day with the heating on and a characteristic peaked profile in the temperature. The analysis is completed by matching this information with the school calendar: in doing so, Sundays are flagged in yellow as warnings, meaning that they are expected anomalies, while the event in red shows an anomalous change of the temperature values; in addition, the *normal daily behaviors* during Holidays vacations are emphasized with the dark yellow line, since they are *anomalous in being normal*. Interpreted within the smart building application context, these anomalies actually point at a potential sub-optimal management of the heating system.

C. Weekly Analyses

The daily normalization does not allow to discern among days with different dynamic range of the temperature, which could instead be revealed by assuming a longer interval normalization. Indeed, normalizing on larger periods would bring in some information about the evolution of the measured temperature over a longer horizon, introducing though issues related to the seasonality fluctuations of the considered signals.

In this sense, the time series analysis on a weekly basis is able to account for the trend over more days by employing an alphabet with a higher cardinality, to increase the description variability, and considering a pattern length of one symbol per hour, to be consistent with the daily descriptor.

The results of this analysis are shown in Fig. 7 where four anomalies are detected. The two weeks from 26/12/2011 to 08/01/2012 are indeed the most anomalous since they present only four temperature peaks, while the standard week shows the six temperature excursions corresponding

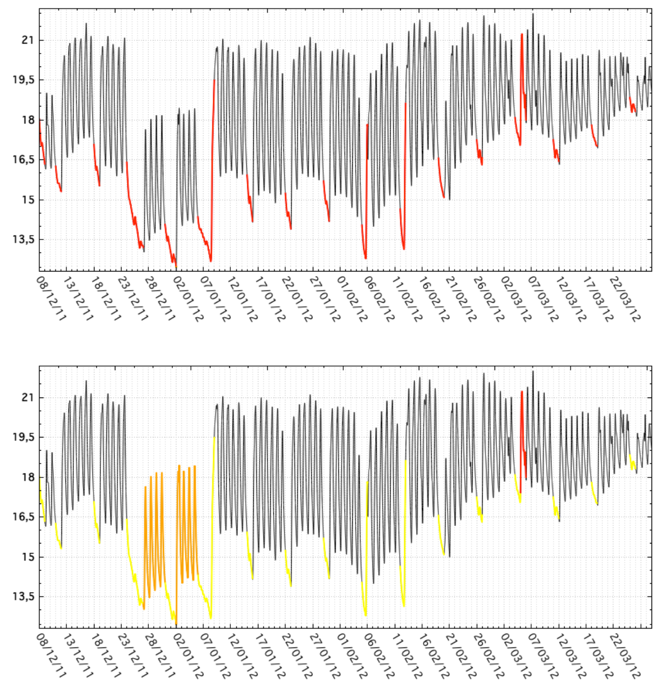

Figure 6. Daily analysis of the school theater temperature data (dates are dd/mm/yyyy) without (top) and with calendar support (bottom).

to the six working days (Monday to Saturday). In addition, the mean temperature value is lower than the normal one, and this behavior can be correlated to the lack of the thermal passive contribution of people's presence in the room, since this period is Season's Holiday. The week from 19/12/2011 to 25/12/2011 is revealed as anomalous because it presents only five daily peaks. Finally, the week from 30/01/2012 to 05/02/2012 presents the standard six day peaks, but it is considered as non-normal because the last day excursion is lower than the others, again for the fact that the room is unoccupied.

From the smart environment application point of view, the first anomaly is actually the most interesting, since it highlights a possible waste of thermal energy in conditioning the rooms unnecessarily: in these situations, making the users aware of a non-correct behavior is the first step for a sustainable use of the resources. Furthermore, it is remarkable how from the analysis of a physical parameter of the environment, the temperature, some information related to the use of the rooms can be derived: the smoothness of the profiles, as well as the lower values, witness that no person is present in the environment to contribute with passive thermal contribution. This information is particularly useful when privacy issues can be raised and there is need to monitor the environment occupancy.

Presented at IEEE SCI 2017
International Workshop on Big Data Applications for Smart Cities (BDASC 2017)

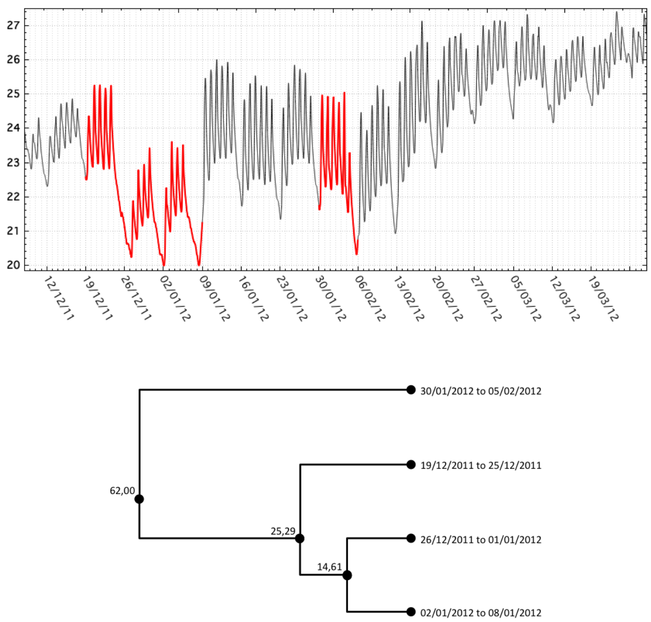

Figure 7. Weekly analysis of one school office temperature data. Top: anomalous week highighted in red. Bottom: dendrogram of the revealed anomalies.

IV. Conclusions and future work

We proposed an approach based on a symbolic representation of the time series and an iterative procedure for model building and anomaly detection. The approach exploits data normalization as a proper element of the modeling, and allows managing complex and noisy situations. The motivating application is the analysis of indoor temperature time series in crowded environments thus casting the data analysis problem in the context of energy awareness and smart buildings.

Together with the detected anomalies, the time series model is obtained as an output of the model building iterations. This model can next be employed *as is* to speed up further analyses with some level of approximation. For example, it could be used for the prediction of the sensor behavior of another data stream obtained from either the same or a different sensor. Combining different models could also lead to the construction of a more complex model derived from several specific sensor time series to study a noisy data series whose training phase would be difficult to perform.

References

[1] I. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “A survey on sensor networks,” *IEEE Comm. Mag.*, vol. 40(8), pp. 102–114, 2002.

[2] K. Romer and F. Mattern, “The design space of wireless sensor networks,” *IEEE Wireless Communications*, vol. 11, no. 6, pp. 54–61, 2004.

[3] T.-C. Fu, “A review on time series data mining,” *Engineering Applications of Artificial Intelligence*, vol. 24, no. 1, pp. 164 – 181, 2011.

[4] V. Chandola, A. Banerjee, and V. Kumar, “Anomaly detection: A survey,” *ACM Comput. Surv.*, vol. 41, no. 3, pp. 15:1–15:58, 2009.

[5] N. A. Lynch, *Distributed Algorithms*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1996.

[6] J. Lin, E. Keogh, S. Lonardi, and B. Chiu, “A symbolic representation of time series, with implications for streaming algorithms,” in *Proc. of the 8th ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery (DMKD)*, 2003, pp. 2–11.

[7] J. Lin, E. Keogh, L. Wei, and S. Lonardi, “Experiencing SAX: A novel symbolic representation of time series,” *Data Min. Knowl. Discov.*, vol. 15, no. 2, pp. 107–144, Oct. 2007.

[8] V. Chandola, A. Banerjee, and V. Kumar, “Anomaly detection for discrete sequences: A survey,” *IEEE Trans. on Knowledge and Data Engineering*, vol. 24, no. 5, pp. 823–839, 2012.

[9] S. Budalakoti, A. Srivastava, and M. Otey, “Anomaly detection and diagnosis algorithms for discrete symbol sequences with applications to airline safety,” *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, vol. 39, no. 1, pp. 101–113, 2009.

[10] E. Keogh, J. Lin, and A. Fu, “Hot SAX: Efficiently finding the most unusual time series subsequence,” in *Proc. of the Fifth IEEE Int. Conf. on Data Mining*, 2005, pp. 226–233.

[11] L. Wei, N. Kumar, V. Lolla, E. J. Keogh, S. Lonardi, and C. Ratanamahatana, “Assumption-free anomaly detection in time series,” in *Proc. 17th Intl. Conf. on Scientific and Statistical Database Management*, 2005, pp. 237–240.

[12] A. Bagnall, J. Lines, A. Bostrom, J. Large, and E. Keogh, “The great time series classification bake off: a review and experimental evaluation of recent algorithmic advances,” *Data Mining and Knowledge Discovery*, pp. 1–55, 2016.

[13] S. Monti, P. Tamayo, J. Mesirov, and T. Golub, “Consensus clustering: A resampling-based method for class discovery and visualization of gene expression microarray data,” *Mach. Learn.*, vol. 52, pp. 91–118, 2003.

[14] T.-Y. Chiu, T.-C. Hsu, and J.-S. Wang, “Ap-based consensus clustering for gene expression time series,” in *Proc. of the 20th Int. Conf. on Pattern Recognition (ICPR)*, 2010, pp. 2512–2515.

[15] E. Keogh, J. Lin, and W. Truppel, “Clustering of time series subsequences is meaningless: Implications for previous and future research,” in *Proc. 3rd IEEE Intl. Conf. on Data Mining*, ser. ICDM ’03, 2003.

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

[17] E. Keogh and S. Kasetty, "On the need for time series data mining benchmarks: A survey and empirical demonstration," *Data Min. Knowl. Discov.*, vol. 7, no. 4, pp. 349–371, Oct. 2003.

[18] K. Chakrabarti, E. Keogh, S. Mehrotra, and M. Pazzani, "Locally adaptive dimensionality reduction for indexing large time series databases," *ACM Trans. Database Syst.*, vol. 27, no. 2, pp. 188–228, 2002.

[19] "Streaming-data algorithms for high-quality clustering," in *Proceedings of the 18th International Conference on Data Engineering*, ser. ICDE '02. Washington, DC, USA: IEEE Computer Society, 2002, pp. 685–. [Online]. Available: http://dl.acm.org/citation.cfm?id=876875.878995