Department of Engineering of Information and Communications

Faculty of Computer Science

University of Murcia

Data Analytics Approaches in IoT based Smart Environments

Ph.D Thesis

Authored by:

Aurora González Vidal

Supervised by:

Dr. Antonio Fernando Skarmeta Gómez

Murica, April 2019

Departamento de Ingeniería de la Información y las Comunicaciones

Facultad de Informática

Universidad de Murcia

Análisis de datos en entornos inteligentes basados en el internet de las cosas

Ph.D Thesis

Presentada por:

Aurora González Vidal

Supervisada por:

Dr. Antonio Fernando Skarmeta Gómez

Murica, Abril 2019

Dedication and acknowledgements

Here goes the dedication.

AUTHOR'S DECLARATION

I declare that the work in this dissertation was carried out in accordance with the requirements of the University's Regulations and Code of Practice for Research Degree Programmes and that it has not been submitted for any other academic award. Except where indicated by specific reference in the text, the work is the candidate's own work. Work done in collaboration with, or with the assistance of, others, is indicated as such. Any views expressed in the dissertation are those of the author.

SIGNED: ..... DATE: .....

TABLE OF CONTENTS

ÍNDICE DE CUADROS

TABLE **Page**

ÍNDICE DE FIGURAS

FIGURE **Page**

Capítulo 1

Resumen

Resumen en español

CHAPTER 2

ABSTRACT

Abstract en inglés. In short, the aim of this thesis is to study and improve every step in the data analytics process which leads to provide better services to the citizens in smart environments, that is, smart cities and smart buildings.

Below, we set out the objectives that must be attained for this aim to be fulfilled, which will serve as a guide to how the thesis is developed.

- O1. Identify and collect datasets relative to smart environments and determine the nature of the data under study
- 02. Find appropriate ways to reduce the volume of such data in order to follow the Big Data paradigm
- 03. Determine the models that better help predict and cluster information regarding energy efficiency
- 04. Develop an IoT platform oriented towards the proper processing, management and analysis of Big volumes of data

CHAPTER 3

INTRODUCTION

Begins a chapter. Example: When the beloved cellist (Christopher Walken - outstanding) of a world-renowned string quartet receives a life-changing diagnosis, the group's future suddenly hangs in the balance: suppressed emotions, competing egos and uncontrollable passions threaten to derail years of friendship and collaboration. Featuring a brilliant ensemble cast (including Philip Seymour Hoffman, Catherine Keener and Mark Ivanir as the three other quartet members), it is a fascinating look into the world of working musicians, and an elegant homage to chamber music and the cultural world of New York. The music, of course, is ravishing (the score is the work of regular David Lynch collaborator Angelo Badalamenti): A Late Quartet hits all the right notes.

3.1 Section

Begins a section.

3.1.1 Subsection

Begins a subsection.

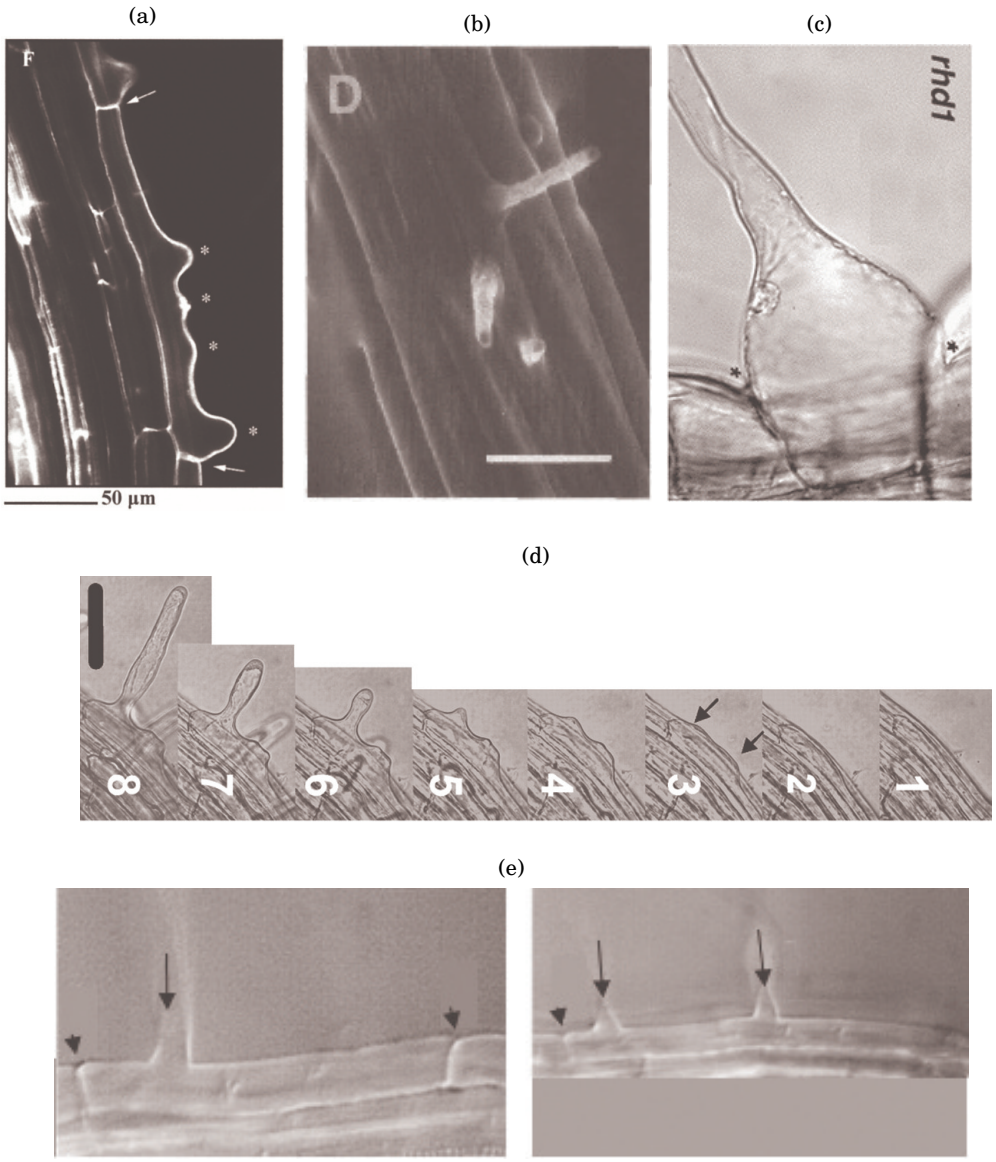


FIGURE 3.1. (a) A mutant RH cell. Asterisks show multiple sites of RH initiation in a single root hair cell (indicated by the arrows). Figure reproduced from [**?**]. (b) Hair-forming cell with three RH initiation locations. The bar represents $50\mu m$. Figure reproduced from [**?**]. (c) Large bump in mutant *rhd1*. Figure reproduced from [**?**]. (d) Mutant overexpressing gene *ROP2*; from right-hand to left-hand, numbers indicate progressive snapshots at different times. RH initiation sites are indicated by the arrows. The bar represents $75\mu m$. Figure reproduced from [**?**]. (e) Mutants affected by auxin. On the left-hand side, RH site is farther away from the apical end (left arrow cap); on the right-hand side, multiple RH locations (arrows). Figure reproduced from [**?**].

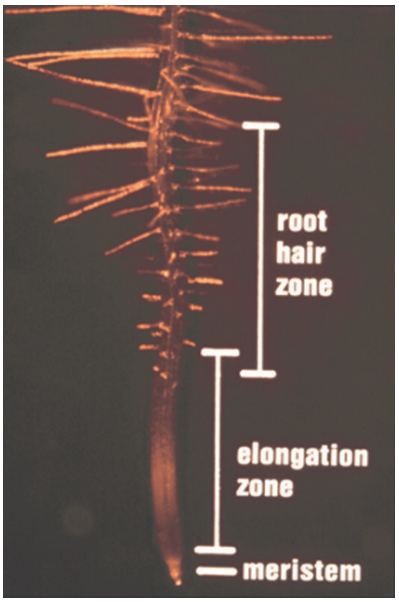


FIGURE 3.2. Developmental zones of an Arabidopsis root. Figure reproduced from [**?**].

CHAPTER 4

PUBLICATIONS COMPOSING THE PHD THESIS

4.1 BEATS: Blocks of Eigenvalues Algorithm for Time Series Segmentation

Title	BEATS: Blocks of Eigenvalues Algorithm for Time Series Segmentation
Authors	Aurora González-Vidal, Payam Barnaghi, and Antonio F. Skarmeta
Type	Journal
Journal	IEEE Transactions on Knowledge and Data Engineering
Impact factor (2017)	2.775
Publisher	IEEE
Volume	30
Year	2018
ISNN	1041-4347 (Print), 1558-2191 (Electronic)
DOI	10.1109/TKDE.2018.2817229
URL	https://ieeexplore.ieee.org/document/8319952/
State	Published
Author's contribution	The author is first author...

BEATS: Blocks of Eigenvalues Algorithm for Time Series Segmentation

Aurora González-Vidal, Payam Barnaghi, *Senior Member, IEEE*, and Antonio F. Skarmeta, *Member, IEEE*

Abstract—The massive collection of data via emerging technologies like the Internet of Things (IoT) requires finding optimal ways to reduce the observations in the time series analysis domain. The IoT time series require aggregation methods that can preserve and represent the key characteristics of the data. In this paper, we propose a segmentation algorithm that adapts to unannounced mutations of the data (i.e., data drifts). The algorithm splits the data streams into blocks and groups them in square matrices, computes the Discrete Cosine Transform (DCT), and quantizes them. The key information is contained in the upper-left part of the resulting matrix. We extract this sub-matrix, compute the modulus of its eigenvalues, and remove duplicates. The algorithm, called BEATS, is designed to tackle dynamic IoT streams, whose distribution changes over time. We implement experiments with six datasets combining real, synthetic, real-world data, and data with drifts. Compared to other segmentation methods like Symbolic Aggregate approXimation (SAX), BEATS shows significant improvements. Trying it with classification and clustering algorithms it provides efficient results. BEATS is an effective mechanism to work with dynamic and multi-variate data, making it suitable for IoT data sources. The datasets, code of the algorithm and the analysis results can be accessed publicly at: https://github.com/auroragonzalez/BEATS.

Index Terms—BEATS, SAX, data analytics, data aggregation, segmentation, DCT, smart cities

1 Introduction

Less than 1 percent of the data that are nowadays captured, stored, and managed by means of the Internet of Things (IoT) and Big Data technologies is being analysed [1]. There exist several challenges in the analysis of data such as high dimensionality, high volume, noise, and data drifts. Data provided by IoT sources (sensory devices and sensing mechanisms) are multi-modal and heterogeneous. Since all of the above mentioned features hinder the execution and generalization of the algorithms, many higher-level representations or abstractions of the raw data have been proposed to address these challenges.

In this paper, we attempt to aggregate and represent large volumes of data in efficient and higher-granularity form. The latter is an attempt to create sequences of patterns and data segments that occur in large-scale IoT data streams. The contribution of our approach is to do such representation on-the-fly since usually data treatment has to be done very quickly, adapting to unpredictable changes in the data or even without prior knowledge.

- *A. González-Vidal and A. F. Skarmeta are with the Department of Information and Communications Engineering, University of Murcia, Murcia 30100, Spain. E-mail: {aurora.gonzalez2, skarmeta}@um.es.*
- *P. Barnaghi is with the Institute for Communication Systems, University of Surrey, Guildford, Surrey GU2 7XH, United Kingdom. E-mail: p.barnaghi@surrey.ac.uk.*

Manuscript received 21 June 2017; revised 21 Jan. 2018; accepted 9 Mar. 2018. Date of publication 0 . 0000; date of current version 0 . 0000.
(Corresponding author: Aurora González-Vidal.)
Recommended for acceptance by E. Terzi.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TKDE.2018.2817229

A use case where large and dynamic datasets are present is smart cities. Data aggregation and pattern representation enables us to find underlying patterns, providing further understanding of *the city data*. Big Data analytics, machine learning and statistical techniques are used to predict, classify and extract information that empowers machines with decision-making capabilities.

IoT data is usually related to physical objects and their surrounding environment. Normally, IoT data is collected together with a timestamp. The collection of several points spaced in time, having a temporal order is known as time series data. Time series can be analysed using various techniques such as clustering, classification and regression (as inputs of models) in the fields of data mining, machine learning, signal processing, communication engineering, and statistics.

Our proposed method is based on splitting time series data into blocks. These blocks can be either overlapping or non-overlapping and they represent subsets of the whole data structure. The method synthesizes independently the information that the blocks contain. It reduces the data points while still preserving their fundamental characteristics (loosing as little information as possible). We propose a novel technique using matrix-based data aggregation, Discrete Cosine Transform (DCT) and eigenvalues characterization of the time series data. The algorithm is called Blocks of Eigenvalues Algorithm for Time series Segmentation (BEATS). We compare BEATS with the state-of-the art segmentation and representation algorithms. We also compare and evaluate the approaches in two of the most common machine learning tasks, classification and clustering, by comparing metrics between each of the transformed datasets. We also present a

use case that is related to smart cities showing the suitability of BEATS for real time data stream analysis. This is shown by explaining how to apply it within a Big Data framework.

The remainder of the paper is organized as follows: Section 2 describes the related work. Section 3 motivates the need of a new approach. Section 4 details the algorithm and briefly explains the mathematical background of the work. Section 5 includes the evaluations in several scenarios using different datasets and a use-case related to smart cities. Section 6 discusses the results of the experiments and Section 7 concludes the paper and describes the future work.

2 Related Work

There are several approaches to represent a numeric time-dependent variable (i.e., a time series). The most basic one is to compute the mean and standard deviation among other statistical measures (e.g., variance, mode). Using those statistics it is not possible to represent all the information that the time series contains. A classical example that supports this claim is the Anscombe's Quartet, [2] that shows how four very different datasets have identical simple statistical properties: mean, variance, correlation and regression coefficients.

In order to reduce the number of data points in a series and create a representation, segmentation methods can be used as a pre-processing step in data analytics.

Definition 1 (Segmentation). *Given a time series T containing n data points, segmentation is defined as the construction of a model $\bar{T}$, from l piecewise segments ($l < n$) such that $\bar{T}$ closely approximates T [3].*

The segmentation algorithms that aim to identify the observation where the probability distribution of a time series changes are called change-point detection algorithms. Sliding windows, bottom-up, and top-down methods are popular change-point detection based approaches. For sliding windows, each segment is grown until it exceeds an error threshold. The next block starts with the new data point not included in the newly approximated segment and so on. In the bottom-up methods, the segments of data are merged until some stopping criteria is met and top-down methods partition the time series recursively until a stopping criteria is met [4].

Another way of classifying the algorithmic methods for segmentation is considering them as online and offline solutions [5]. While offline segmentation is used when the entire time series is previously given, the online segmentation deals with points that arrive at each time interval. In offline mode, the algorithm first learns how to perform a particular task and then it is used to do it automatically. After the learning phase is completed, the system cannot improve or change (unless we consider incremental learning or retraining). On the other hand, online algorithms can adapt to possible changes in the environment. Those changes are known as "drifts". Whereas top-down and bottom-up methods can only be used offline, sliding windows are applicable to both circumstances.

After segmentation, the representation of the time series based on the reduction can be regarded as an initial step that reduces the load and improves the performance of tasks such as classification and clustering. The use of such algorithms can be generally regarded in two ways:

- Representation methods: Extracting features from the whole time series or its segments and applying machine learning algorithms (Support Vector Machines, Random Forest, etc) in order to classify them or compute the distance between the time series representation for clustering.
- Instanced based methods (similarities): Computing the distance matrix between the whole series and using it for clustering or classification applying a k-nearest neighbour approach [6] by finding the most similar (in distance) time series in the training set.

BEATS is based on the first perspective since as stated in *Bagnall et al The greatest improvement can be found through choice of data transformation, rather than classification algorithm* [7]. However, we review the work made using both approaches since the ultimate goal of our time series representation is to make the time series data more aggregated and better represented for further processing.

2.1 Whole Series Similarities

Similarity measures are used to quantify the distance between two raw time series. The list of approaches is vast and the comparison between well-known methods has lead to the conclusion that the benchmark for classification is dynamic time warping (DTW) since other techniques proposed before 2007 were found not significantly better [8].

Similar results have been stated in [9] when comparing DTW with more recent distance measures as: Weighted DTW [10], Time warp edit (TWE) [11] and Move-split-merge (MSM) [12] together with a slight accuracy improvement (1 percent) when using Complexity invariant distance (CID) [13] and Derivative transform distance (DTD_C) [14].

When computation time is not a problem, the best approach is to use a combination of nearest neighbour (NN) classifiers that use whole series elastic distance measures in the time domain and with first order derivatives: Elastic ensemble (EE) [15]. However, if a single measure is required a choice between DTW and MSM is recommended, with MSM preferred because of its overall performance.

In the clustering domain, the number of evaluated similarity distances is even higher, due to the nature of the problem. An extensive description of similarity measures can be found in [16]. DTW and CID are also used in clustering the raw time series [17], [18].

2.2 Intervals

Various algorithms focus on deriving features from intervals of each series. For a series of length m, there are $m(m-1)/2$ possible contiguous intervals.

Piecewise Linear Representation (PLR) [19] methods are based on the approximation of each segment in the form of straight lines and include the perceptually important points (PIP), Piecewise Aggregate Approximation (PAA) [20], and the turning point (TP) method [21].

The state-of-the-art models Time Series Forest (TSF) [22] and Learned pattern similarity (LPS) [23] generate many different random intervals and classifiers on each of them, ensembling the resulting predictions.

TSF trains several trees in a random forest fashion but each tree uses as data input the $3\sqrt{m}$ statistics features (mean, standard deviation and slope) of the $\sqrt{m}$ randomly selected intervals.

LPS can be regarded as an approximation of an autocorrelation function. For each series, they generate a random number l of series by randomly selecting a fixed number w of elements of the primitive one. A column of the generated $l * n \times w$ matrix is chosen as the class and a regression tree is built (autocorrelation part). After that, for every series the number of rows of the matrix (originated by the raw series) that reside in each leaf node is counted. Concatenating these counts the final representation of the series is formed. Then, a 1-NN classifier is applied to process the time series data.

2.3 Symbolic Aggregate Approximation (SAX)

Among all the techniques that have been used to reduce the number of points of a time series data, SAX has specially attracted the attention of the researchers in the field. SAX has been used to asses different problems such as finding time series discords [24], finding motifs in a database of shapes [25], and to compress data before finding abnormal deviations [26] and it has repeatedly been enhanced [27], [28], [29].

SAX allows a time series of length n to be reduced to a string of length l ($l < n$). The algorithm has two parameters: window length w and alphabet size α, and it involves three main steps [30]:

- Normalization: standardizes the data in order to have a zero mean and a standard deviation of one;
- Piecewise Aggregation Approximation (PAA): divides the original data into the desired number of windows and calculates the average of data falling into each window; and
- Symbolization: discretizes the aggregated data using an alphabet set with the size represented as an integer parameter α, where $\alpha > 2$.

  As normalized time series data assumes a Gaussian distribution for the data, the discretization phase allows to obtain a symbolic representation of the data by mapping the PAA coefficients to a set of equiprobable breakpoints that are produced according to the alphabet size α. The breakpoints determine equal-sized areas under the Gaussian curve [31] in which each area is assigned to an alphabet character.

Since SAX representation does not consider the segment trends, different segments with similar average values may be mapped to the same symbols. Among the multiple enhancements done to SAX (see related work section of [28] and [29]) we highlight the following works:

- Extended SAX (ESAX) [27]: adds maximum and minimum along with the original SAX representation.
- SAX Trend Distance (SAX_TD) [28]: defines the trend distance quantitatively by using the starting and ending point of the segment and improved the original SAX distance with the weighted trend distance.
- SAX with Standard Deviation (SAX_{SD}) [29]: adds the standard deviation of the segment to its SAX representation.

The Vector Space Model (VSM) is combined with SAX in [32] in order to discover and rank time series patterns by their importance to the class. Similarly to shapelets, SAX-VSM looks for time series subsequences which are characteristic representatives of a class. The algorithm converts all training time series into bags of SAX words and uses *tf-idf* weighting and cosine similarity in order to rank by importance the subsequences of SAX words according to the classes.

2.4 Shapelets

Shapelets are subsequences of time series that identify with the class that the time series belongs to.

The Fast shapelets (FS) [33] algorithm discretises and approximates shapelets using SAX. The dimensionality of the SAX dictionary is reduced through masking randomly selected letters (random projection).

Learned shapelets (LS) [34] optimizes a classification loss in order to learn shapelets whose minimal euclidean distances to the time series are used as features for a logistic regression model. An improvement of such model is the use of DTW instead of euclidean distance [35].

The Fused LAsso Generalized eigenvector method (FLAG) [36] is a combination of the state-of-the-art feature extraction technique of generalized eigenvector with the fused LASSO that reformulates the shapelet discovery task as a numerical optimization problem instead of a combinatorial search.

Finally, we take into consideration the clustering algorithm k-shape [37], a centroid-based clustering algorithm that can preserve the shapes of time-series sequences. They capture the shape-based similarity by using a normalized version of the cross-correlations measure and claims to be the only scalable method that significantly outperforms k-means.

2.5 Ensembles

So far we have reviewed how data transformation techniques are applied to different algorithms in order to improve their accuracy and to reduce the computation time. COTE algorithm [38] uses a collective of ensembles of classifiers on different data transformations.

The ensembling approach in COTE is unusual because it adopts a heterogeneous ensemble rather than resampling schemes with weak learners. COTE contains classifiers constructed in the time, frequency, change (autocorrelations), and shapelet transformation domains (35 in total) combined in alternative ensemble structures. Each classifier is assigned a weight based on the cross validation training accuracy, and new data are classified with a weighted vote.

The results of evaluations in COTE show that the simple collective formed by including all classifiers in one ensemble is significantly more accurate than any of its components.

3 Motivation and Contributions

As it can be seen among the segmentation techniques that we referenced in section 2, we have mentioned not only the representation techniques but also how the whole classification and clustering procedure is performed by combining representation with machine learning algorithms. We intended to show that our representation method is an efficient alternative segmentation method to be employed in time series data processing.

One commonality of the several studies that we have reviewed is that most of the existing algorithms use normalization that re-scales the data.

However, there are few studies that do not apply re-scaling and normalization. BEATS uses a non-normalized algorithm for constructing the segment representation.

The concept *drift* appears when a model built in the past is no longer fully applicable to the current data. Concept drift is due to a change in the data distribution according to a single feature, to a combination of features or in the class boundaries, since the underlying source generating the data is not stationary.

The potential changes in the data might happen in:

- The prior probability P(y_i);
- The conditional probability P($x|y_i$);
- The posterior probability P($y_i|x$); and
- A combination of the above.

Where x is the predicted data and y_i is the observed data.

These changes can cause two kinds of concept drift: real and virtual [39].

If only the data distribution changes without any effect on the output, i.e., changes in P(y_i) and/or P($x|y_i$) that does not affect P($y_i|x$), it is called virtual drift.

When the output, i.e., P($y_i|x$), also changes it is called real concept drift.

In the IoT domain and especially in smart city data analysis, we are interested in the second type of drift which will be referred as *data drift* in this paper [40]. Some examples where a data drift may occur in smart cities are related to the replacement of sensors (different calibration), sensor wear and tear [41] or drastic changes to the topics of discussion in social media used for crowdsensing [42].

There are several existing methods and solution addressing the concept drift for supervised learning [41], and some recent ones also for unsupervised learning [40]. However, we focus on the initial step of the analysis (i.e., pre-processing). We claim that not only the model has to be adaptive but also the way that we segment the inputs has to take into account the dynamics of the data and be able to efficiently deal with the changes in the structure of the data.

A considerable challenge in segmentation is to find a common way to represent the data. This is due to the variety of ways to formulate the problem in terms of defining the key parameters (number of segments, segmentation starting point, length of segments, error function, user-specified threshold, etc.).

The first step in SAX algorithm is assuming that for a particular problem that we deal with, the data follows a normal distribution or at least we have a sufficiently large number of samples in order to say that the distribution of the data is approximately normal, appealing to the central limit theorem [43]. Nevertheless, this is a strong assumption because there are many scenarios in which this might not be the case; for example:

- Outliers and noise: data from physical devices usually contains noise and outliers that affect the identification of the correct parameters of the distribution.
- Data follows different distribution.
- Fast data: two of the V's from the 7V's Big Data challenges [44] are *velocity* and *variety*. Traditionally in data mining, batch data is processed in an offline manner using historical data. However, in IoT applications we need to consider short-term snapshots of the data which are collected very quickly. Thus, we need adaptive methods that catch up with the changes during their operation.

All mentioned algorithms lack of at least one of such 3 problems too. We have developed an algorithm that does not require normalization of the data. The latter will also help to preserve the value of the data points (i.e., magnitude of the data). The lack of sensitivity to magnitude in the algorithms that make assumptions about the normalized distribution and use Z-normalization makes them less efficient in analysing correlation and regression. Another requirement is the application of the algorithm in an online way and using sliding windows. Nonetheless, we have to be able to compute the distance between the aggregated time series. Considering these requirements we have designed the BEATS algorithm.

4 BEATS PRESENTATION

This section describes our proposed algorithm and discusses its mathematical and analytical background. We present BEATS and show the effect of each step of the algorithm in a block of data.

4.1 BEATS Construction

Transforms, in particular integral transforms, are used to reduce the complexity in mathematical problems. In order to decorrelate the time features and reveal the hidden structure of the time series, they are transformed from the time domain into other domains. Well-known transformations are the Fourier Transform, which decomposes a signal into its frequency components, and the Karhunen-Loeve Transform (KLT) which decorrelates a signal sequence.

Discrete Cosine Transform (DCT) is similar to Discrete Fourier Transform (DFT) but uses cosines obtained from the discretization of the kernel of the Fourier Transform. DCT transfers the series to the frequency domain. Among the four different cosine transformations classified by Wang [45], the second one (i.e., DCT-II) is regarded as one of the best tools in digital signal processing [46] (times series can be regarded as a particular case of signals). Due to its mathematical properties such as unitarity, scaling in time, shift in time, the difference property, and the convolution property, DCT-II is asymptotically equivalent to the KLT where under certain (and general) conditions KLT is an optimal but impractical tool to represent a given random function in the mean square error sense (MSE). KLT is said to be an optimal transform because:

- It completely decorrelates the signal in the transform domain;
- It minimizes the MSE in bandwidth reduction or data compression;
- It contains the most variance (energy) in the fewest number of transform coefficients; and
- It minimizes the total representation entropy of the sequence.

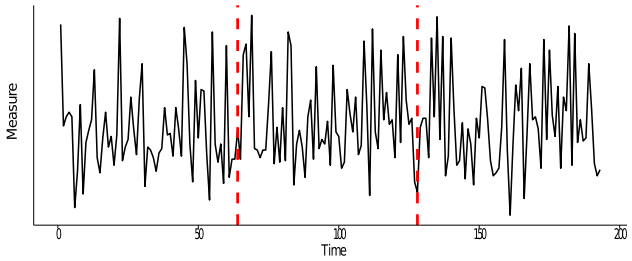


Fig. 1. An example of a time series divided into blocks of 64 observations.

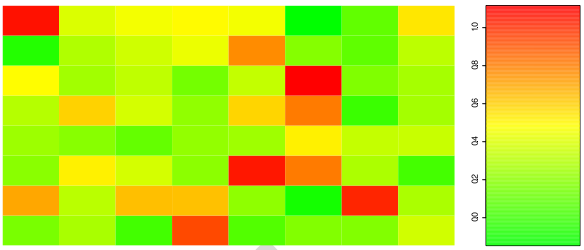


Fig. 2. The heatmap of the matrix obtained from the first block of time series data.

The details of the proof of the above statements can be found in [46]. Understanding the properties of the DCT, we use it to transform our time series data.

We apply the transformation essentially by using the compression of a stream of square 8x8 blocks, taking reference from the standards in image compression [47] where DCT is widely used (e.g., JPEG). Since 8 is a power of 2, it will ease the performance of the algorithm.

As an illustration, we provide an example. We have divided the time series shown in Fig. 1 as blocks of 64 observations that are shown using a dashed red line. If we arrange the first block row-wise into a squared matrix M, we can visualize that the information is spread through the matrix as the heatmap shown in Fig. 2.

It should be noted that while our raw time series data is represented in value/time, a 2D transformation is applied to the data. This is based on the assumption that in each block, the neighbour values of a selected observation m_{ij} (eg. $m_{i-1j}, m_{ij-1}, m_{i-1j-1}$ are correlated. In time series with very rapid changes in the data, small block sizes will be more suitable and if the changes are not very rapid size block can be larger. In this paper, we use a common 8×8 block size for our description.

Intuitively, each 8×8 block includes 64 observations of a discrete signal which is a function of a two-dimensional (2D) space. The DCT decomposes this signal into 64 orthogonal basis signals. Each DCT coefficient contains one of the 64 unique *spatial frequencies* which comprise the *spectrum* of the input series. The DCT coefficient values can be regarded as the relative amount of the spatial frequencies contained in the 64 observations [47].

Let M be the 8×8 input matrix. Then, the transformed matrix is computed as $D = UMU^{\mathsf{T}}$, where U is an 8×8 DCT matrix. U coefficients for the $n \times n$ case are computed as shown in Eq. 1:

$$U_{ij} = \begin{cases} \frac{\sqrt{2}}{2} & i,j = 1 \\ \cos\left(\frac{\pi}{n}(i-1)(j-\frac{1}{2})\right) & i,j > 1. \end{cases} \quad (1)$$

The formula of Eq. (1) is obtained using Eq. (5) (Appendix 8). Finally, we multiply the first term by $\frac{1}{\sqrt{2}}$ in order to make the DCT-II matrix orthogonal. After applying DCT, the information is accumulated in its upper-left part, as it is shown in the heatmap in Fig. 3.

Each of the 64 entries of the matrix D is quantized by pointwise division of the matrices D and Z, where the elements of the quantization matrix Z are integer values ranging from 1 to 255.

Quantization is the process of reducing the number of bits needed to store an integer value by reducing the precision of the integer. Given a matrix of DCT coefficients, we can divide them by their corresponding quantizer step size and round it up depending on its magnitude, normally 2 decimals. If the maximum of the DCT matrix is small, the number of decimals is selected by the operation $||\lfloor \log_{10} max \rfloor - 4|$, where $\lfloor \log_{10} max \rfloor$ returns the position of the first significant figure of the maximum number in the transformed matrix D. This step is used to remove the high frequencies or to discard information which is not very significant in large-scale observations.

The selected matrix Z is the standard quantization matrix for DCT [48].

After the quantization process, a large number of zeroes appears in the bottom-right position of the matrix $Q = \frac{D}{Z}$, i.e., it is a sparse matrix.

We extract the 4×4 upper-left matrix that contains the information of our 64 raw data and compute the eigenvalues, which in our case are: $0.18605, 0.02455, 0.00275 + 0.00843i, 0.00275 - 0.00843i$.

Using BEATS so far we have significantly reduced the number of points of our time series from 64 to 4 but we have also converted its components into complex numbers. These complex numbers (eigenvalues vector) represent the original block in a lower dimension. This eigenvalues vector is used in BEATS to represent the segments and hence, it is the potential input for the machine learning models. However, it is not always possible to feed machine learning algorithms with complex numbers and the eigenvalues could be complex numbers. To solve this problem, we compute the modulus of the eigenvalues and remove the repeated ones (they are presented in pairs so the information would be repeated).

In case that there are no complex numbers in the output of BEATS, we will conserve the first three values, since the latter values are sorted in a descending order. This means that we have represented the original 64 observations as

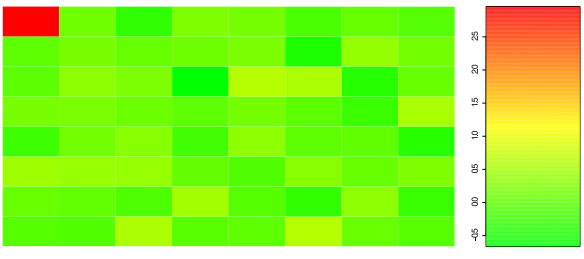


Fig. 3. The heatmap of the DCT matrix.

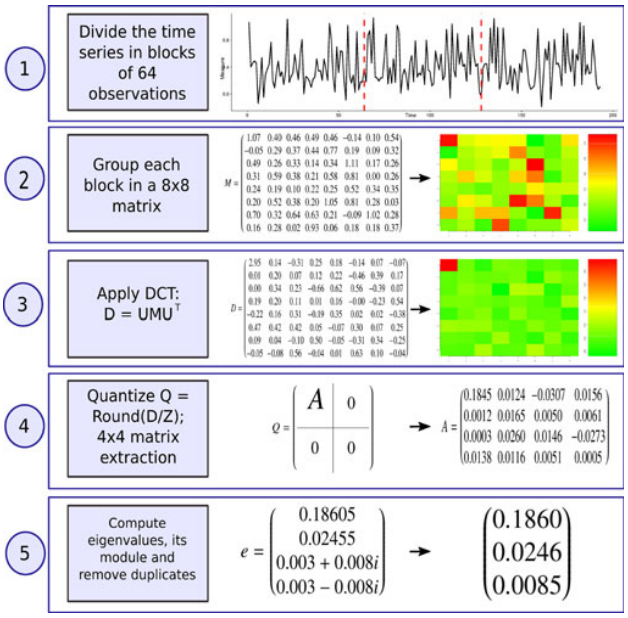


Fig. 4. BEATS is shown step by step with an example.

three values. In our example, the final representation (modulus of the eigenvalues) consists of $0.1860, 0.0246, 0.0085$.

The BEATS process is summarized in Fig. 4.

We also consider the relevance of the direct computation of the eigenvalues of the 8×8 matrix M in order to assure that the DCT and its quantization contribute to the aggregation of the information. We refer to this method throughout the paper as Eigen.

4.2 Complexity Analysis of BEATS

The time complexity is represented as a function of the input time series size (n). Regarding the different steps of BEATS, the processes that have a key impact on the run time are DCT, which is a double matrix multiplication, i.e., $O(n^3)$; pointwise matrix division for the quantization, i.e., $O(n^2)$ and eigenvalue computation, i.e., $O(\beta^3)$, where n is the size of the matrix block (square root of the amount of data that compounds each block), and β ($\leq n$) is the size of the extracted matrix from which we compute the eigenvalues. Although we have set the values to $n = 8$ and $\beta = 4$, we compute the complexity in general terms.

So far, the dominant task regarding the complexity is the DCT function. For about the past 40 years, many fast algorithms have been reported to enhance the computation of discrete cosine transforms [49]. In order to improve the efficiency of the algorithm, we have implemented a popular way of computing the DCT of our N-points time series. We use a 2N-points Fast Fourier Transform (FFT). This has reduced the complexity to $O(n^2 log(n))$ [50].

Hence, for each block we have a complexity of $O(n^2 log(n) + \beta^3)$. Let N be the size of our time series data; if we do not use sliding windows, we will apply the algorithm $\frac{N}{n \times n}$ times, so the complexity is $\frac{N}{n \times n} O(n^2 log(n) + \beta^3)$. As we can see, the complexity of the algorithm grows linearly depending on the number of blocks where we have to apply the computations.

By applying multiple processing architectures, the complexity problem nowadays can also depend on how efficiently we can parallelize the processing load. Parallelising the BEATS algorithm is very simple since the computations are *block dependent* and no information out of the block is required for each individual calculation. This makes the process ideal to be done using graphics processing units (GPUs), and thereby minimising the latency of the computation.

5 Experimental Evaluation

We perform two data mining processes: classification and clustering. Following our approach the data is going to be transformed by the two methods: BEATS and Eigen, summarized as follows:

- BEATS: 8×8 matrix blocks of the data, discrete cosine transformation, and quantization of each of the matrices, reduction to a 4×4 matrix, removal of the duplicated modulus of the complex eigenvalues and selection of the first three values.
- Eigen: 8×8 matrix blocks of the data, computation of the eigenvalues of the matrices, removal of the duplicated modulus of the complex eigenvalues, and selection of the first three values.

Having introduced several algorithms in Section 2, we compare BEATS and Eigen with common existing state-of-the-art methods that show an improvement in comparison with the primitive ones.

The algorithms' code has been accessed from the authors' public repositories when available. When not, R software and Python have been used in order to program them.

We perform each of the techniques using several datasets in order to analyse the type of problems that our algorithm performs better than other methods. It is possible to use sliding windows for our method. In the experiment, we consider a slide of 8 observations. The evaluations also include a cross validation step in order to find their parameters.

A smart cities use case where we cluster traffic data is also presented. The intention is to see how BEATS is suitable for different scenarios including online smart cities applications.

5.1 Datasets

We give a short explanation of the datasets that are used to evaluate the algorithm. Four of the datasets are obtained from the UCR Time Series Classification Archive [51]: ArrowHeads, Coffee, FordA, Lightning7 and ProximalPhalanxOutlineAgeGroup. For each dataset we use, when provided, the train sample in order to find the hyperparameters of the model and then, we test their classification performance with the test set. For clustering we use only the training set. When the split is not provided, which is the case in one of the datasets (the randomly generated by us), we use 75 percent of the samples for the training set and 25 percent of the samples for testing.

The datasets that are used in the experiments are briefly described below.

Arrow Heads (Real and Without Drifts). The Arrow Heads dataset[1] contains 211 series having 192 observations classified into three different classes. The arrowhead data consists

1. http://www.cs.ucr.edu/~eamonn/time_series_data/

of outlines of the images of arrowheads [52]. The shapes of the projectile points are converted into a time series using the angle-based method and they are classified based on shape distinctions such as the presence and location of a notch in the arrow. The classification of projectile points is an important topic in anthropology. According to our method, we reduced the dataset to 72 observations.

Lightning7 (Real and Long). We use the Lightning7 dataset that gathers data related to transient electromagnetic events associated with the lightning natural phenomenon. Data is gathered with a satellite with a sample rate of 800 microseconds and a transformation is applied in order to produce series of length 637.

The classes of interest are related to the way that the lightning is produced.[2]

Initially, each measurement (time series) carries 320 variables. Using our method, we have reduced the dataset to 96 variables.

Random LHS Generator Lift (Synthetic and with Drifts). A dataset with data drifts is also used in our experiments. In this case, we have evaluated the algorithms with the data generated by using the code from the Repository[3] described in [53], which was first used in [40]. The drift is introduced both by shifting the centroids in randomized intervals and by changing the data distribution function used to randomly draw the data from the centroids that are selected through Latin Hypercube Sampling (LHS). This dataset is created for smart cities data analysis and allows to create sample datasets that simulate dynamic and multi-variate data streams in a smart environment. The data generator is developed in the context of the CityPulse smart city project.[4]

The number of centroids is set to ten and we generated 300 series that follow three different distributions (triangular, Gaussian and exponential). Initially, each set (time series) carries 192 variables. Using our method, we reduced the dataset to 51 variables.

Coffee (Real-World Data). The Coffee dataset[1] contains 56 series having 286 observations classified into two different classes. The Coffee data consists of the series generated by the Fourier transform infrared spectroscopy of two species of coffee: Arabica and Robusta. Originally, such method intended to serve as an alternative to wet chemical methods for authentication and quantification of coffee products [54]. Using BEATS, we reduced the dataset to 57 observations which represent the patterns that occur in the dataset. This can be used for further analysis and classification of coffee types.

FordA (Real-World Data). The FordA dataset[1] contains 4921 series having 500 observations each classified into two different classes. The data was generated on the context of a classification competition. The problem is to diagnose whether a certain symptom exists in a automotive subsystem using the engine noise as a measurement. Both training and test data set were collected in typical operating conditions, with minimal noise contamination. Using BEATS, we reduced the dataset to 100 observations. The BEATS observations are more resilient to noise and provide an efficient way to discover and extract patterns from real-world raw data.

ProximalPhalanxOutlineAgeGroup (Real-World Data from Images). The ProximalPhalanxOutlineAgeGroup dataset[1] contains 605 series having 80 observations each classified into three different classes. The dataset was created [55] for testing the efficacy of hand and bone outline detection and whether these outlines could be helpful in bone age prediction. The problem involves using the outline of one of the phalanges of the hand in order to predict whether the subject is one of three age groups . Using BEATS, we reduced the dataset to 9 observations per subject. This observations provide a reduced feature set that ease the analysis tasks.

5.2 Classification

Classification of time series analysis is a classic problem consisting of building a model based on labelled time series data and using the model to predict the label of unlabelled time series samples.

The applications of this technique are widely extended in many areas, ranging from epilepsy diagnosis based on time series recorded by electroencephalography devices (electrical activity generated by brain structures over the scalp) [56] to uncovering customers' behavior in the telecommunication industry [57], and predicting traffic patterns in a smart city environment.

After transforming our data using BEATS and Eigen, we followed the general data modelling process proposed in [58] to classify the series: standarization, splitting the dataset into training and test sets, choosing the model, selecting the best hyperparameters of each model using 10-fold cross validation on the training set and checking the accuracy of the model using the test set. With respect to the methodology followed in [58], we improve the way of looking for the hyperparemeters of the algorithms using the python package optunity since it contains various optimizers for hyperparameter tuning.

Among other options like grid search, random search and genetic algorithms, we have chosen particle swarm implementation since it is shown to surpass the performance of other solutions [59].

The models that we use to combine with BEATS and Eigen are the widely known Random Forest (RF) and Support Vector Machines (SVM) with Radial Basis Function Kernel.

Whereas Random Forest deals with *small n large p-problems*, high-order interactions and correlated predictor variables, SVMs are more effective for relatively small datasets with fewer outliers. Generally speaking, Random Forests may require more data. Both of the algorithm show better performance when combined with SVM.

The tuning of SVM has been done without deciding the kernel in advance. That means, the kernel (linear, polynomial or RBF) is considered as an hyperparameter.

According to the discussion in Section 2, we compare our method with:

- Original time series (i.e., raw data): DTW with 1-NN classification since, after many trials, it is still the benchmark of comparison for distance based classification. Having a complexity of $O(n^2)$ that under

2. http://www.timeseriesclassification.com/description.php?Dataset=Lightning7

3. https://github.com/auroragonzalez/BEATS/tree/master/data/random_LHS_generator_drift

4. http://www.ict-citypulse.eu

TABLE 1
Accuracy of Each Method Using as Inputs Each of the Segmented Time Series

Model \ dataset	Arrow Heads	Lightning7	Random Generator	Coffee	Ford A	Proximal
BEATS-SVM	**0.81**	0.7	**0.75**	**1**	**0.75**	**0.85**
Eigen-SVM	0.79	0.72	0.73	**1**	0.74	0.8
DTW-1NN	0.67	0.75	0.71	0.87	0.66	0.81
SAX-VSM	0.68	0.59	0.52	0.96	0.09*	0.75
TSF	0.73	0.75	**0.75**	0.97	**0.75**	0.85
FLAG	0.57	0.76	0.67	**1**	0.73	0.64
COTE	0.78	**0.8**	0.7	**1**	**0.75**	0.83

**The bag of words generated by a wide majority of the test subjects is not related to the ones generated by the train step. This implies that their TF*IDF weights are not computed and it is not possible to compute the cosine similarity. In consequence, the method is not valid for many of the cases, producing the reported bad results.*

certain circunstances [60] could be reduced to $O(n)$ using lower bounds such as LB_{Keogh} or $LB_{Improved}$ [61].

- Intervals: We choose TSF in order to make the comparison since it is more modern and quicker than the rest.
  Its complexity is $O(t * m * n * logn)$, where t = number of trees and m = number of splits or segments.
- Symbolic approximations: In the classification task, we use SAX-VSM. The complexity is linear: $O(n)$.
- Shapelets: FLAG is the newest, the quickest and claims to be better than its predecessors.
  Its complexity is $O(n^3)$.
- Ensembles: COTE. It is an ensemble of dozens of core classifiers many of which having a quadratic, cubic or even bi-quadratic complexity. It is the most computationally expensive in this list.

The results are shown in Table 1. It is important to mention that not only accuracy results but also the time that it takes the algorithm to run both training and test phases including input transformation, has improved. This runtime is shown in Fig. 5, where a logarithmic transformation is applied to the data in order to improve visibility.

We have depicted both metrics: accuracy and running time in a plot that summarises the results over all the datasets. Both metrics have been scaled per dataset and we have computed the average performance per model that is represented by the bigger points in the plot.

In order to make a more consistent analysis of the results, we have generated 100 Random LHS Generator Lift datasets and the model accuracy of the models using violin plots (see Fig. 6), which together with the regular statistics that boxplot provide they show the probability density of the data at different values of accuracies. While the differences between BEATS-SVM, TSF and COTE are not statistically significant (p-value = 0.7 $>$ 0.05), BEATS-SVM is very quick in comparison to COTE and that BEATS is also more versatile than the rest since it can be combined with any classification algorithms.

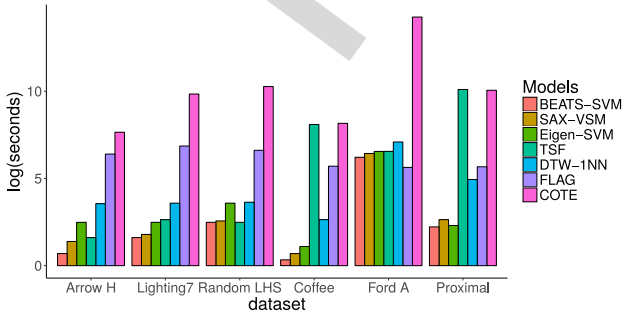


Fig. 5. Running time (log(sec)) and programming language of the algorithms.

5.3 Clustering

Clustering is used to identify the structure of an unlabeled dataset by organising the data into homogeneous groups where within-group-object similarity is minimized and between-group-object dissimilarity is maximized. The process is done without consulting known class labels. Clustering is an unsupervised machine learning method. In particular, time series clustering partitions time series data into groups based on similarity or distance; so that time series data in the same cluster are similar.

Clustering has tackled tasks such as the assignment of genes with similar expression trajectories to the same group [62]. The creation of profiles of the trips carried out by tram users [63] or the acquisition of energy consumption predictions by clustering houses [64] are among examples of using clustering methods.

After transforming our data using BEATS and Eigen, we applied the connectivity based algorithm *hierarchical agglomerative clustering* and the centroid based algorithm *k-means* to

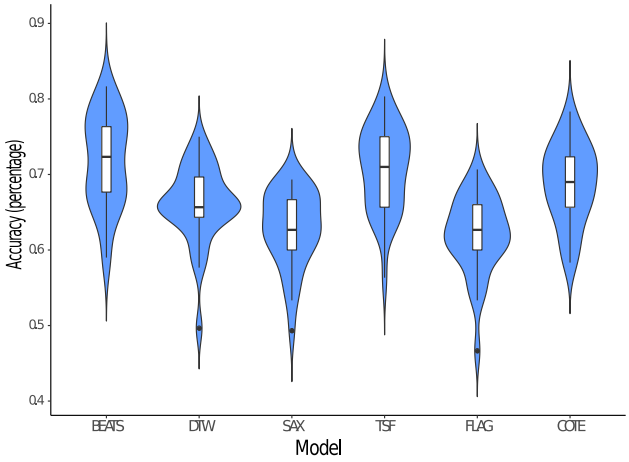


Fig. 6. Classification accuracy on the 100 randomly generated datasets.

TABLE 2
Silhouette Coefficient of Each Method Using as Inputs Each of the Segmented Time Series

Model \ dataset	Arrow Heads	Lightning7	Random Generator	Coffee	Ford A	Proximal
BEATS-HC	**0.6**	0.25	**0.45**	0.25	**0.46**	0.4
Eigen-HC	0.58	**0.31**	0.25	0.26	0.36	0.38
DTW	0.33	0.21	0.44	0.21	0.12	0.31
SAX$_{SD}$- HC	0.53	0.06	0.19	0.13	0	0.33
k-shape	0.44	0.19	0.05	**0.43**	0.38	**0.5**

cluster the time series datasets. In the hierarchical clustering, the selected agglomerative method is *complete linkage*, meaning that the distance between two clusters is the maximum distance between their individual components (in each time series). Hierarchical clustering seems to be a better partner for both of them.

The dissimilarity matrix contains the distances between the pairs of time series. We use the cosine dissimilarity for the rest of the segmentations (BEATS and Eigen). The cosine dissimilarity is calculated as one minus the cosine of the included angle between elements of the time series (see Eq. (2))

$$\text{dissimilarity} = 1 - \frac{\mathbf{XY}}{\|\mathbf{X}\|\|\mathbf{Y}\|} = 1 - \frac{\sum_{i=1}^{n} X_i Y_i}{\sqrt{\sum_{i=1}^{n} X_i^2}\sqrt{\sum_{i=1}^{n} Y_i^2}}. \quad (2)$$

Finally, for both methods we have used a fixed number of clusters. As we were aware of the classification groups (our data is labeled), we applied the algorithms setting apriori the number of clusters k and used the silhouette coefficient as a metric for measuring the cluster quality.

The silhouette coefficient is an internal measure that combines the measurement of cohesion and separation. Cluster cohesion measures how closely related the objects in a cluster are. Cluster separation measures how well separated the clusters are from each other. The silhouette coefficient for a subject i is defined as

$$s(i) = \frac{b(i) - a(i)}{max\{a(i), b(i)\}}, \quad (3)$$

where $a(i)$ is the average distance between i and each of the points of the assigned cluster and $b(i)$ is the average distance between i and each of the points of the next best cluster. This value can be used to compare the quality of different cluster results.

From the definition it is clear that $s(i) \in [-1, 1]$. Meanwhile a silhouette coefficient value closer to 1 means that the clustering is good; a value close to -1 represents less efficiency in the categorization for the clusters. When it is close to 0, it means that the point is in the border between two clusters.

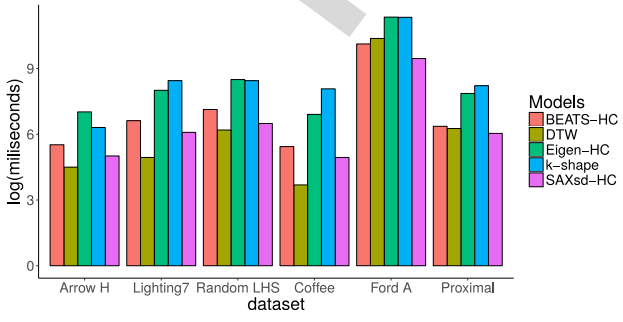


Fig. 7. Running time (log(milisec)) of the clustering algorithms.

According to the discussion in Section 2 we will analyse:

- Original time series: DTW distance using the tight lower bound of [61], that makes it faster.
- Symbolic approximations: We have taken the most modern improvement that SAX has experienced: SAX$_{SD}$. The MINDIST function that returns the minimum distance between the original time series of two words [65] is enhanced with the distance between the standard deviation of each segment.
- Shapelets: k-shape is the model chosen in this direction.

The results of the clustering experiments done in the training sets are shown in Table 2. The run time of the algorithms is shown in Fig. 7. In this case, all the algorithms have been coded using the same programming language so we consider that the graph is enough in order to estimate the different algorithms complexity regarding time.

5.4 Big Data Use Case: Traffic in Smart Cities

In this section we apply BEATS in a smart cities related use-case: traffic data clustering, done in an online and distributed way.

5.4.1 *BEATS Implementation for Big Data*

In contrast to the traditional analysis procedure where data is first stored and then processed in order to deploy models, the major potential of the data generated by IoT is accomplished by the realization of continuous analytics that allow to make decisions in real time.

There are three types of data processing: Batch Processing, Stream Processing and Hybrid Processing.

Batch processing operates over a group of transactions collected over a period of time and reports results only when all computations are done, whereas stream processing produces incremental results as soon as they are ready [66].

Regarding the available Big Data Tools, we have considered Hadoop[5] and Spark[6] Big Data frameworks. Hadoop was designed for batch processing. All data is loaded into HDFS and then MapReduce starts a batch job to process that data. If the data changes the job needs to be ran again. It is step by step processing that can be paused or interrupted, but not changed.

Apache Spark allows to perform analytical tasks on distributed computing clusters. Sparks real-time data

5. http://hadoop.apache.org/
6. https://spark.apache.org/

processing capability provides substantial lead over Hadoops MapReduce and it is essential for online time series segmentation and representation.

The Spark abstraction for a continuous stream of data is called a Discretized Stream or DStream . A DStream is a micro-batch of Resilient Distributed Datasets, RDDs. That means, a DStream is represented as a sequence of RDDs. RDDs are distributed collections that can be operated in parallel by arbitrary functions and by transformations over a sliding window of data (windowed computations).

5.4.2 BEATS Adapted to Spark Technology

For the online implementation of BEATS we have decided to use pyspark, the Spark Python API that exposes the Spark programming model to Python.

There are many works proposing online time series processing but few of them that have implemented it. In [67] is highlighted that MapReduce is not the appropiate techonology for rolling window time series prediction and proposes a index pool data structure.

Pyspark allows us to use the Spark Streaming functionalities that are needed in order to implement BEATS online. In Section 3 we have seen that BEATS algorithm can be separately applied to windows of the data. Therefore we associate the data received within one window to one RDD, that can be processed in a parallel way.

A suitable type of RDDs for our implementation is key/value pairs. In detail, the key is an identifier of the time series (e.g., sensor name) and the value is the sequence of values of our time series that fall in the window. That way the blocks are exposed to operations that give the possibility to act on each key in parallel or regroup data across the network.

The transformations that we use are:

- Window: use for creating sliding window of time over the incoming data.
- GroupByKey: grouping the incoming values of the sliding window by key (for example, same sensor data).
- Map: The Map function applied in parallel to every pair (key, value), where the key is the time series, values are a vector and the function depends on what has to be done.

5.4.3 The Applied Scenario

We use one of the real-world datasets obtained from the collection of datasets of vehicle traffic in the City of Aarhus in Denmark for a period of 6 months.[7] The dataset is provided in the context of the CityPulse smart city project.

The selected dataset gathers 16971 samples of data from sensors situated in lamp posts covering an area around 2345m.[8] The variables considered for the analysis are: flow (numbers of cars between two points) and average speed. Each variable is a time series.

In order to simulate an online application we consider that the BEATS segmentation is carried out on hourly based data.

7. http://iot.ee.surrey.ac.uk:8080/datasets.html#traffic
8. http://iot.ee.surrey.ac.uk:8080/datasets/traffic/traffic_june_sep/index.html

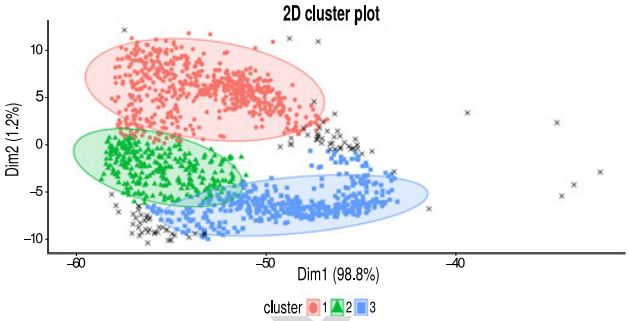


Fig. 8. Plot of the DBSCAN clusters using traffic data.

To achieve this, since the data is collected every 5 mins., a sliding window of size 12 is selected. The goal of the clustering is to determine the status of the road in terms of the traffic flow and occurrences. For every window of 128 observations (64 for each variable) BEATS obtains three flow related representatives and three speed related representatives.

Each observation of the final input dataset for the clustering model represents one window of the raw data. The final dataset has 6 variables and 1409 samples. This means a reduction of around 75 percent of data.

The data is gathered by anonymously collecting Wi-Fi and Bluetooth signals transmitted by travelers' smartphones or in-vehicle systems. This infrastructure provides noisy data in cases such as stopped vehicles in traffic jam, buses with a lot of passengers.

In order to tackle the presence of outliers and noise, the selected clustering technique is density-based spatial clustering (DBSCAN). DBSCAN groups points that are closely packed together. Points that do not fit into any of the main groups because they lie in low-density regions are marked as outliers. The hyper-parameters of DBSCAN are minimum number of points required to form a dense region (MinP) and ϵ in order to find the ϵ-neighborhood of each point. We set that clusters contain at least a 20 percent of the data and $\epsilon = 4.014$. Using such configuration, we obtain 3 different clusters and a 8 percent of data that cannot be classified in any of the previous, i.e., outliers. The description of the clusters, including the number of points n that belong to each of the clusters and the mean μ and standard deviation sd for both flow and speed is:

- Cluster 1 (n = 618): High flow (μ = 30.97, sd= 12.66) and medium speed (μ = 102.5, sd= 10.2);
- Cluster 2 (n = 271): Medium flow (μ = 15.97, sd= 8.4) and high speed (μ = 110, sd= 9.21); and
- Cluster 3 (n = 432): Low flow (μ = 6.1, sd= 5.56) and low/medium speed (μ = 97.8, sd= 14.3).

In order to represent the data in lower dimension, we select the first two principal components of the data using Principal Components Analysis (PCA). The obtained clusters are shown in Fig. 8. Crosses in black colour represent the noise data. We have also projected the clusters in the three flow related components of BEATS, so that clusters can be visualized in a 3D form as presented in Fig. 9.

Regarding this application, we can conclude that clustering methods applied to the segments generated by BEATS are able to characterise the status of the roads by grouping the values in an effective form.

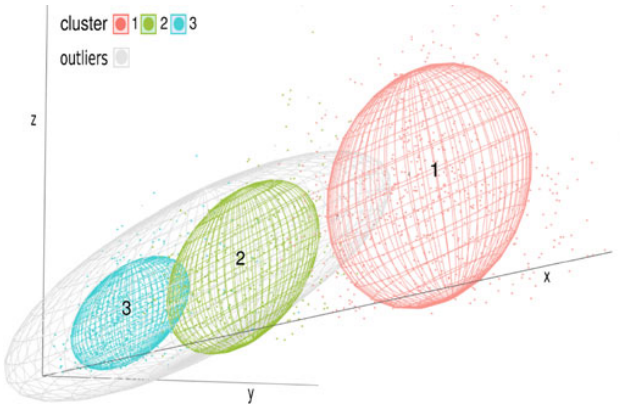


Fig. 9. 3D plot of the DBSCAN clusters using traffic data.

Using a computer with an Intel i5 Processor, 8GB RAM Memory, Ubuntu 16.04 operative system and the statistical software R 3.4.3 [68], the running time of DBSCAN using BEATS segmented data is 0.25 seconds. However, to run the DBSCAN with raw data it takes around 35 seconds. The later confirms again the suitability of BEATS in current IoT scenarios.

6 Discussion

As we have described in the paper, the randomness and predictability of a real-world time series changes over time due to several factors.

The existing solutions for pattern creation and abstraction in time-series data often work based on statistical measures (which have limited representation and granularity), symbolic methods such as SAX (which assumes that the data is normally distributed and requires normalization of the data), or signal processing and stream processing methods such as wavelet or Fourier transforms (which act as filters and can extract features from the data but do not provide a pattern representation/abstraction).

Our proposed model combines a series of methods to create a window based abstraction of time series data and uses a frequency domain function combined with characteristic value measures that represents the overall direction of the dataframe (i.e., an n-dimensional matrix constructed during our windowing/slicing process) as a vector.

BEATS is an algorithm that process data streams whose randomness and predictability varies depending on the segment of data. The proposed algorithm is useful specially in applications such as smart cities where results of the segmentation and processing algorithms are used in order to make fast decisions regarding traffic, energy, light regulation, etc. This can be made by combining various sensory data and other historical data. In general terms, the intention is to predict and manage what is occurring in order to provide informed or automated decisions for repetitive tasks that can be handled by machines. BEATS offers a powerful solution to aggregate and represent large-scale streaming data in a quick and adaptable way. It uses blocks of eigenvalues in a much lower-dimensionality (with a high aggregation rate) which preserves the main information and characteristics of the data. Since BEATS uses eigenvalues, it provides a homogeneous way to represent multi-modal and heterogeneous streaming data. In other words, all different types of numerical streaming data are transformed into vectors of eigenvalues that, in principal, preserve and represent the magnitude and overall direction of the data in a lower-dimensionality space. This not only allows to compare and combine different blocks of data from various data streams, but also provides a unified way to represent the blocks of data as patterns in the form of eigenvalues.

In this paper, we mainly target a key step after collection of the data: aggregation. Aggregation of data becomes a very significant task in order to extract the key characteristics of the data in lower-dimensionality. We segment the time series and make a reduction for each time series at a rate of 60 $\sim$ 70 percent when using overlapping windows. The independence between blocks that our algorithm provides is one of its most important features. BEATS also presents other qualities such as adapting to drifts and low latency.

BEATS reduces the data by using the eigenvalues of a submatrix of the DCT transformation. These eigenvalues represent the key-characteristics of the data.

The evaluation is performed using classification and clustering, two of the classical machine learning tasks using several types of datasets. The inputs of the models are the different representations introduced in the paper: BEATS and Eigen together with raw data for the other models.

Classification is measured by accuracy. This allows us to perform a test for equality of proportions, that is a χ^2 test of independence in order to assure that the differences between accuracies are statistically significant.

For the Arrow Heads dataset we find that BEATS combined with SVM outperforms all the algorithms. However, the differences between COTE and BEATS are not statistically significant ($\chi^2(1) = 0.37$, p-value = $0.54 > 0.05$). On the other hand, the difference between TSF and BEATS are statistically significant ($\chi^2(1) = 4.8$, p-value = $0.04 < 0.05$).

In the case of Lightning7, there are several models that outperform BEATS. The winning one is COTE. Nonetheless, COTE is very complicated, time demanding and computationally expensive. The rest only overperforms BEATS by 6 percent at most.

In the case of Random LHS Generator Lift, TSF and BEATS perform similarly.

In the Coffee dataset, we observe that several approaches (including BEATS) achieve a 100 percent accuracy on classification.

In FordA, BEATS, TSF and COTE perform similarly. However, BEATS is the quickest amongst them.

Finally, in the Proximal dataset TSF and BEATS perform similarly in terms of accuracy. However, BEATS is again quicker.

Even though COTE and TSF are strong rivals to BEATS, it should be noted that the computation time and simplicity of BEATS makes it useful to use in rapid analysis having still good results. Also, due to its nature is very adaptable and easy to combine with any other classification algorithm different than SVM.

The clustering experiment is evaluated by comparing the hundredths of the silhouette coefficients, where each hundredth is going to be counted as *a point* in the below description.

BEATS is 7 points above SAX_{SD} for the Arrow Heads dataset, 1 point above *DTW* in the Random LHS Generator Lift set and 8 points above k-shape in Ford A. Being the most computationally expensive of all the clustering algorithms under study, as it can be seen in Fig. 7, k-shape outperforms BEATS in two datasets: Coffee and Proximal.

It can be said that in clustering, BEATS behaves better when we are using long datasets since it outperforms every algorithms in both metrics: silhouette coefficient and running time in the biggest dataset: *FordA*.

Finally, by applying DBSCAN to cluster traffic data, we noticed that BEATS performs efficiently since the clusters represent different situations of the use-case in terms of traffic flow and speed.

7 Conclusions and Future Work

In this paper, we introduce a novel algorithm called BEATS, which aggregates and represents time series data in blocks of lower-dimensional vectors of eigenvalues. BEATS is not sample dependent so it adapts to data drifts in the underlying data streams.

The BEATS abstractions can be combined with various machine learning models to discover patterns, identify correlations (within or between data streams), extract insights and identify activities from the data. In this paper, we have used several datasets and have shown several use cases that demonstrate how the BEATS abstractions can be used for clustering, analysis and grouping the activities and patterns in time-series data.

Compared to existing segmentation methods, BEATS shows significant improvements in representing datasets with drifts. When combined with classification and clustering methods, we have shown that it can obtain competitive results compared with other state-of-the-art but more complex and time consuming methods.

For the BEATS algorithm evaluation we have fixed the length of the segments at 64; so the only parameter to take into consideration was the slide of the window, that we have kept constantly equal to 8, so the blocks of transformed data intersect. Nevertheless, the optimization of the sliding window is an open issue to be addressed in future work.

For the clustering tasks, it is important to take into account that the definition of similarity is subjective. The similarity depends on the domain of application.

By using BEATS, we are able to restructure the streaming data in a 2D way and then transform it into the frequency domain using DCT. The algorithm finds a smaller sequence that contains the key information of the initial representative. This aggregation provides an opportunity to eliminate repetitive content and similarities that can be found in the sequence of data.

The eigenvalues vectors are a homogeneous representation of the data streams in BEATS that allow us to go one step further in understanding of the sequences and patterns that can be considered as the data structure of a data series in an application domain (e.g., smart cities).

Its applications can be extended to several other domains and various patterns/activity monitoring and detection methods. The future work will focus on applying 3D cosine transform and adaptive block size estimation.

Appendix A

Definition A.1 (Integral transform). *The integral transform of the funtion $f(t)$ with respect to the kernel $K(t,s)$ is*

$$F(t) = \int_{-\infty}^{\infty} K(t,s) f(t) dt, \quad (4)$$

if the integral exists.

The kernel of the Fourier Transformation is $K(t,s) = e^{-its}$, and, in particular for the cosine fourier transformation $K(t,\omega) = cos(t,\omega)$. If we discretize the kernel we can reach that $K_c(j,k) = cos(\frac{jk\pi}{N})$, where N is an integer.

Definition A.2. (Discrete Cosine Transformation (DCT) - II). *DCT is a linear and invertible function*

$$f : \mathbb{R}^n \longrightarrow \mathbb{R}^n$$

where $\mathbb{R}$ denotes the set of real numbers or, equivalently, on a $n \times n$ matrix, defined by:

$$f_j = \sum_{k=0}^{n-1} cos\left(\frac{\pi}{n} j \left(k + \frac{1}{2}\right)\right) \text{ where } j = 0, 1, \ldots, n-1 \quad (5)$$

Acknowledgments

This work has been partially funded by MINECO grant BES-2015-071956, PERSEIDES TIN2017-86885-R project, and ERDF funds, by the European Comission through the H2020-ENTROPY-649849 EU Project, and the H2020 FIESTA Project under grant agreement no. CNECT-ICT-643943.

References

[1] J. Gantz and D. Reinsel, "The digital universe in 2020: Big data, bigger digital shadows, and biggest growth in the far east," *IDC iView: IDC Analyze Future*, vol. 2007, pp. 1–16, 2012.

[2] D. Abbott, *Applied Predictive Analytics: Principles and Techniques for the Professional Data Analyst*. Hoboken, NJ, USA: Wiley, 2014.

[3] E. J. Keogh and M. J. Pazzani, "An enhanced representation of time series which allows fast and accurate classification, clustering and relevance feedback," in *Proc. 4th Int. Conf. Knowl. Discovery Data Mining*, vol. 98, pp. 239–243, 1998.

[4] E. Keogh, S. Chu, D. Hart, and M. Pazzani, "Segmenting time series: A survey and novel approach," *Data Mining Time Series Databases*, vol. 57, pp. 1–22, 2004.

[5] H. Aksoy, A. Gedikli, N. E. Unal, and A. Kehagias, "Fast segmentation algorithms for long hydrometeorological time series," *Hydrological Processes*, vol. 22, no. 23, pp. 4600–4608, 2008.

[6] X. Xi, E. Keogh, C. Shelton, L. Wei, and C. A. Ratanamahatana, "Fast time series classification using numerosity reduction," in *Proc. 23rd Int. Conf. Mach. Learn.*, 2006, pp. 1033–1040.

[7] A. Bagnall, L. Davis, J. Hills, and J. Lines, "Transformation based ensembles for time series classification," in *Proc. SIAM Int. Conf. Data Mining*, 2012, pp. 307–318.

[8] X. Wang, A. Mueen, H. Ding, G. Trajcevski, P. Scheuermann, and E. Keogh, "Experimental comparison of representation methods and distance measures for time series data," *Data Mining Knowl. Discovery*, vol. 26, pp. 1–35, 2013.

[9] 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 Knowl. Discovery*, Springer, vol. 31, no. 3, pp. 606–660, 2017.

[10] Y.-S. Jeong, M. K. Jeong, and O. A. Omitaomu, "Weighted dynamic time warping for time series classification," *Pattern Recognit.*, vol. 44, no. 9, pp. 2231–2240, 2011.

[11] P.-F. Marteau, "Time warp edit distance with stiffness adjustment for time series matching," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 2, pp. 306–318, Feb. 2009.

[12] A. Stefan, V. Athitsos, and G. Das, "The move-split-merge metric for time series," *IEEE Trans. Knowl. Data Eng.*, vol. 25, no. 6, pp. 1425–1438, Jun. 2013.

[13] G. E. Batista, X. Wang, and E. J. Keogh, "A complexity-invariant distance measure for time series," in *Proc. SIAM Int. Conf. Data Mining*, 2011, pp. 699–710.

[14] T. Górecki and M. Łuczak, "Non-isometric transforms in time series classification using DTW," *Knowl.-Based Syst.*, vol. 61, pp. 98–108, 2014.

[15] J. Lines and A. Bagnall, "Time series classification with ensembles of elastic distance measures," *Data Mining Knowl. Discovery*, vol. 29, no. 3, pp. 565–592, 2015.

[16] S. Aghabozorgi, A. S. Shirkhorshidi, and T. Y. Wah, "Time-series clustering–a decade review," *Inf. Syst.*, vol. 53, pp. 16–38, 2015.

[17] V. Hautamaki, P. Nykanen, and P. Franti, "Time-series clustering by approximate prototypes," in *Proc. 19th Int. Conf. Pattern Recognit*, 2008, pp. 1–4.

[18] G. E. Batista, E. J. Keogh, O. M. Tataw, and V. M. De Souza, "Cid: An efficient complexity-invariant distance for time series," *Data Mining Knowl. Discovery*, vol. 28, no. 3, pp. 634–669, 2014.

[19] Y. Zhu, D. Wu, and S. Li, "A piecewise linear representation method of time series based on feature points," in *Proc. Int. Conf. Knowl.-Based Intell. Inf. Eng. Syst.*, 2007, pp. 1066–1072.

[20] E. J. Keogh and M. J. Pazzani, "A simple dimensionality reduction technique for fast similarity search in large time series databases," in *Proc. Pacific-Asia Conf. Knowledge. Discovery Data Mining*, 2000, pp. 122–133.

[21] N. T. Nguyen, B. Trawiński, R. Katarzyniak, and G.-S. Jo, *Advanced Methods for Computational Collective Intelligence*. Berlin, Germany: Springer, 2012, vol. 457.

[22] H. Deng, G. Runger, E. Tuv, and M. Vladimir, "A time series forest for classification and feature extraction," *Inf. Sci.*, vol. 239, pp. 142–153, 2013.

[23] M. G. Baydogan and G. Runger, "Time series representation and similarity based on local autopatterns," *Data Mining Knowl. Discovery*, vol. 30, no. 2, pp. 476–509, 2016.

[24] E. Keogh, J. Lin, and A. Fu, "Hot sax: Efficiently finding the most unusual time series subsequence," in *Proc. 5th IEEE Int. Conf. Data Mining*, 2005, pp. 8–pp.

[25] X. Xi, E. J. Keogh, L. Wei, and A. Mafra-Neto, "Finding motifs in a database of shapes," in *Proc. Int. Conf. Data Mining*, 2007, pp. 249–260.

[26] C. D. Stylios and V. Kreinovich, "Symbolic aggregate approximation (SAX) under interval uncertainty," in *Proc. Annu. Conf. North Amer. Fuzzy Inf. Process. Soc. held Jointly 5th World Conf. Soft Comput.. IEEE*, 2015, pp. 1–7.

[27] B. Lkhagva, Y. Suzuki, and K. Kawagoe, "Extended SAX: Extension of symbolic aggregate approximation for financial time series data representation," in *Proc. Data Eng. Workshop*, vol. 4A-i8, 2006.

[28] Y. Sun, J. Li, J. Liu, B. Sun, and C. Chow, "An improvement of symbolic aggregate approximation distance measure for time series," *Neurocomputing*, vol. 138, pp. 189–198, 2014.

[29] C. T. Zan and H. Yamana, "An improved symbolic aggregate approximation distance measure based on its statistical features," in *Proc. 18th Int. Conf. Inf. Integr. Web-Based Appl. Serv.*, 2016, pp. 72–80.

[30] J. Lin, E. Keogh, L. Wei, and S. Lonardi, "Experiencing sax: A novel symbolic representation of time series," *Data Mining Knowl. Discovery*, vol. 15, no. 2, pp. 107–144, 2007.

[31] S. Kolozali, D. Puschmann, M. Bermudez-Edo, and P. Barnaghi, "On the effect of adaptive and non-adaptive analysis of time-series sensory data," *IEEE Internet Things J.*, vol. 3, no. 6, pp. 1084–1098, 2016.

[32] P. Senin and S. Malinchik, "Sax-VSM: Interpretable time series classification using sax and vector space model," in *Proc. IEEE 13th Int. Conf. Data Mining*, 2013, pp. 1175–1180.

[33] T. Rakthanmanon and E. Keogh, "Fast shapelets: A scalable algorithm for discovering time series shapelets," in *Proc. SIAM Int. Conf. Data Mining*, 2013, pp. 668–676.

[34] J. Grabocka, N. Schilling, M. Wistuba, and L. Schmidt-Thieme, "Learning time-series shapelets," in *Proc. 20th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2014, pp. 392–401.

[35] M. Shah, J. Grabocka, N. Schilling, M. Wistuba, and L. Schmidt-Thieme, "Learning DTW-shapelets for time-series classification," in *Proc. 3rd IKDD Conf. Data Sci.*, 2016, Art. no. 3.

[36] L. Hou, J. T. Kwok, and J. M. Zurada, "Efficient learning of timeseries shapelets," in *Proc. 13th AAAI Conf. Artif. Intell.*, 2016, pp. 1209–1215.

[37] J. Paparrizos and L. Gravano, "k-shape: Efficient and accurate clustering of time series," in *Proc. ACM SIGMOD Int. Conf. Manag. Data*, 2015, pp. 1855–1870.

[38] A. Bagnall, J. Lines, J. Hills, and A. Bostrom, "Time-series classification with COTE: The collective of transformation-based ensembles," *IEEE Trans. Knowl. Data Eng.*, vol. 27, no. 9, pp. 2522–2535, Sep. 2015.

[39] M. Sayed-Mouchaweh, *Learning from Data Streams in Dynamic Environments*. Berlin, Germany: Springer, 2016.

[40] D. Puschmann, P. Barnaghi, and R. Tafazolli, "Adaptive clustering for dynamic iot data streams," *IEEE Internet Things J.*, vol. 4, no. 1, pp. 64–74, Feb. 2017.

[41] J. Gama, I. Žliobaitė, A. Bifet, M. Pechenizkiy, and A. Bouchachia, "A survey on concept drift adaptation," *ACM Comput. Surveys* [Online]. Available: http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=6703726, vol. 46, no. 4, 2014, Art. no. 44.

[42] C. Lifna and M. Vijayalakshmi, "Identifying concept-drift in twitter streams," *Procedia Comput. Sci.*, vol. 45, pp. 86–94, 2015.

[43] C. M. Grinstead and J. L. Snell, *Introduction to Probability*. Providence, RI, USA: American Mathematical Society, 2012.

[44] M. Ali-ud-din Khan, M. F. Uddin, and N. Gupta, "Seven v's of big data understanding big data to extract value," in *Proc. Zone 1 Conf. Amer. Soc. Eng. Edu.*, 2014, pp. 1–5.

[45] Z. Wang, "Fast algorithms for the discrete W transform and for the discrete fourier transform," *IEEE Trans. Acoust. Speech Signal Process.*, vol. 32, no. 4, pp. 803–816, Aug. 1984.

[46] K. R. Rao and P. Yip, *Discrete Cosine Transform: Algorithms, Advantages, Applications*. Cambridge, MA, USA: Academic Press, 2014.

[47] G. K. Wallace, "The jpeg still picture compression standard," *IEEE Trans. Consumer Electron.*, vol. 38, no. 1, pp. xviii–xxxiv, Feb. 1992.

[48] A. C. Bovik, *The Essential Guide to Image Processing*. Cambridge, MA, USA: Academic Press, 2009.

[49] G. Bi and Y. Zeng, *Transforms and Fast Algorithms for Signal Analysis and Representations*. Berlin, Germany: Springer, 2004.

[50] J. Makhoul, "A fast cosine transform in one and two dimensions," *IEEE Trans. Acoust. Speech Signal Process.*, vol. 28, no. 1, pp. 27–34, Feb. 1980.

[51] Y. Chen, E. Keogh, B. Hu, N. Begum, A. Bagnall, A. Mueen, and G. Batista, "The UCR time series classification archive," Jul. 2015, [Online]. Available: www.cs.ucr.edu/ eamonn/time_series_data/

[52] L. Ye and E. Keogh, "Time series shapelets: A new primitive for data mining," in *Proc. 15th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2009, pp. 947–956.

[53] D. Puschmann, "Random lhs generator drift," 2016. [Online]. Available: https://github.com/UniSurreyIoT/random-LHS-generator-drift

[54] R. Briandet, E. K. Kemsley, and R. H. Wilson, "Discrimination of arabica and robusta in instant coffee by fourier transform infrared spectroscopy and chemometrics," *J. Agricultural Food Chemistry*, vol. 44, no. 1, pp. 170–174, 1996.

[55] L. M. Davis, "Predictive modelling of bone ageing," Ph.D. dissertation, University of East Anglia, Norwich, Norfolk, 2013.

[56] A. Anguera, J. Barreiro, J. Lara, and D. Lizcano, "Applying data mining techniques to medical time series: An empirical case study in electroencephalography and stabilometry," *Comput. Structural Biotechnology J.*, vol. 14, pp. 185–199, 2016.

[57] Y. Yang, Q. Yang, W. Lu, J. Pan, R. Pan, C. Lu, L. Li, and Z. Qin, "Preprocessing time series data for classification with application to crm," in *Proc. Australasian Joint Conf. Artifi. Intell.*, 2005, pp. 133–142.

[58] A. González-Vidal, V. Moreno-Cano, F. Terroso-Sáenz, and A. F. Skarmeta, "Towards energy efficiency smart buildings models based on intelligent data analytics," *Procedia Comput. Sci.*, vol. 83, pp. 994–999, 2016.

[59] S.-W. Lin, K.-C. Ying, S.-C. Chen, and Z.-J. Lee, "Particle swarm optimization for parameter determination and feature selection of support vector machines," *Expert Syst. Appl.*, vol. 35, no. 4, pp. 1817–1824, 2008.

[60] C. A. Ratanamahatana and E. Keogh, "Three myths about dynamic time warping data mining," in *Proc. SIAM Int. Conf. Data Mining*, 2005, pp. 506–510.

[61] D. Lemire, "Faster retrieval with a two-pass dynamic-time-warping lower bound," *Pattern Recognit.*, vol. 42, no. 9, pp. 2169–2180, 2009.

[62] Z. Bar-Joseph, A. Gitter, and I. Simon, "Studying and modelling dynamic biological processes using time-series gene expression data," *Nature Rev. Genetics*, vol. 13, no. 8, pp. 552–564, 2012.

[63] M. V. Moreno, F. Terroso-Saenz, A. Gonzalez, M. Valdes-Vela, A. F. Skarmeta, M. A. Zamora-Izquierdo, and V. Chang, "Applicability of big data techniques to smart cities deployments," *IEEE Trans. Ind. Informat.*, vol. 13, no. 2, pp. 800–809, Apr. 2017.
[64] C. Costa and M. Y. Santos, "Improving cities sustainability through the use of data mining in a context of big city data," in *Proc. Int. Conf. Data Mining Knowl. Eng.*, 2015, vol. 1, pp. 320–325.
[65] J. Lin, E. Keogh, S. Lonardi, and B. Chiu, "A symbolic representation of time series, with implications for streaming algorithms," in *Proc. 8th ACM SIGMOD Workshop Res. Issues Data Mining Knowledge Discovery*, 2003, pp. 2–11.
[66] M. Hirzel, H. Andrade, B. Gedik, G. Jacques-Silva, R. Khandekar, V. Kumar, M. Mendell, H. Nasgaard, S. Schneider, R. Soulé, et al., "Ibm streams processing language: Analyzing big data in motion," *IBM J. Res. Develop.*, vol. 57, no. 3/4, pp. 7–1, 2013.
[67] L. Li, F. Noorian, D. J. Moss, and P. H. Leong, "Rolling window time series prediction using MapReduce," in *Proc. IEEE 15th Int. Conf. Inf. Reuse Integr.*, 2014, pp. 757–764.
[68] R Core Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, 2016. [Online]. Available: https://www.R-project.org/

Aurora González Vidal received the graduated degree in mathematics from the University of Murcia, in 2014. In 2015, she received a fellowship to work in the Statistical Division of the Research Support Services, where she specialized in statistics and data analyisis. Since 2015, she has been working toward the PhD degree in computer science, focusing her research on data analytics for energy efficiency and studied for a master's degree in Big Data. She was a visiting PhD student at the University of Surrey, where she worked on the study of segmentation of time series.

Payam Barnaghi is a reader in machine intelligence in the Institute for Communication Systems Research with the University of Surrey. He was the coordinator of the EU FP7 CityPulse project on smart cities. His research interests include machine learning, the Internet of Things, the Semantic Web, adaptive algorithms, and information search and retrieval. He is a senior member of the IEEE and a fellow of the Higher Education Academy.

Antonio F. Skarmeta received the BS (Hons.) degree in computer science from the University of Murcia, the MS degree in computer science from the University of Granada, Spain, and the PhD degree in computer science from the University of Murcia. He is a full professor with the Department of Information and Communications Engineering, University of Murcia. He is involved in numerous projects, both European and National. Research interests include mobile communications, artificial intelligence, and home automation. He is a member of the IEEE.

4.2 Applicability of Big Data Techniques to Smart Cities Deployments

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2016.2605581, IEEE Transactions on Industrial Informatics

Applicability of Big Data Techniques to Smart Cities Deployments

M. Victoria Moreno, Fernando Terroso-Sáenz, Aurora González-Vidal, Mercedes Valdés-Vela
Antonio F. Skarmeta, Miguel A. Zamora and Victor Chang

***Abstract*—This paper presents the main foundations of Big Data applied to Smart Cities. A general Internet of Things based architecture is proposed to be applied to different smart cities applications. We describe two scenarios of big data analysis. One of them illustrates some services implemented in the smart campus of the University of Murcia. The second one is focused on a tram service scenario where thousands of transit-card transactions should be processed. Results obtained from both scenarios show the potential of the applicability of this kind of techniques to provide profitable services of smart cities, such as the management of the energy consumption and comfort in smart buildings, and the detection of travel profiles in smart transport.**

***Index Terms*—Internet of Things; Smart City; Big Data; Predictive Models; Transit-card Mining**

I. Introduction

A Smart City emerges when the urban infrastructure is evolved through the Information and Communication Technologies (ICT) [1]. The paradigm of Internet of Things (IoT) [2] has enabled the emergence of a high number of different communication protocols, which can be used to communicate with commercial devices using different data representations. In this context, it is necessary an IoT-based platform to manage all interoperability aspects and enable the integration of optimal Artificial Intelligence (AI) techniques in order to model contextual relationships.

In urban environments there is a huge amount of different data sources. Plenty of sensors are distributed around cities, most of them installed in indoor spaces. This situation has brought new analytics mechanisms and tools that provide insight allowing us to have an effective and collaborative way to operate the machines [3]. Furthermore, there are numerous mobile data sources like smart phones, smart-cards, wearable sensors and, in the case of vehicles, on-board sensors. All these sensors provide information that makes possible to detect urban dynamic patterns. Nonetheless, most existing management systems of cities are not able to utilize fully and effectively this vast amount of data and, as a result, there is large volumes of data which is not exploited. In this direction, many AI techniques in Computer Science have been introduced to deal with the processing of huge amount of data to extract useful information (or termed by knowledge) from data [4], this trend is known as Big Data.

This paper is intended to analyze the interest of big data for smart cities. In order to face the above-mentioned aspects we propose a general architecture for smart city applications, which is modelled in four layers with different functionalities. Then, we show some applications of big data analysis in two scenarios, both dealing with sensed data coming from both static and dynamic sources. Among other objectives, the first scenario intends to create a distributed framework to share large volumes of heterogeneous information for their use in smart building applications. In this work we focus on presenting the deployments and implementations carried out in smart buildings to achieve energy efficiency. For this, different problems like indoor localization, thermal comfort characterization and energy consumption modelling have been solved through the application of big data techniques. The second example is centered on the public tram service in the City of Murcia (Spain), looking for giving insight into the great amount of data generated by the service's transit cards. In this scenario, big data techniques are applied to extract mobility patterns in public transport.

Hence, this paper faces up three aspects of nowadays smart cities which need to be solved, and for each one of them we provide some research contributions through the application of convenient big data techniques. These contributions are:

- The design and instantiation of an IoT-based architecture for applications of smart cities.
- The approach of an efficient management of energy in smart buildings.
- The extension of the data analysis for detection of urban patterns which can be used to improve public transport applied to the public tram service.

The structure of this paper is as follows: Section II enumerates the challenges that current smart cities still have to face, and proposes a general IoT-based architecture for smart-city services which is modeled in layers. Section III describes a first application of smart city where big data techniques have been applied to get energy efficiency in the buildings of a Smart Campus. Section IV presents a second smart city application that addresses the urban pattern recognitions in public transport. Section V summarizes the main benefits of applying big data techniques to the two scenarios of smart cities addressed in this paper. Finally, Section VI gives some conclusions and an outlook of future work.

M. Victoria Moreno, Fernando Terroso-Sáenz, Aurora González-Vidal, Mercedes Valdés-Vela, Antonio F. Skarmeta and Miguel A. Zamora are with the Department of Information and Communications Engineering, University of Murcia, 30100 Spain, e-mail: (mvmoreno, fterroso, auroragonzalez2, mdvaldes, skarmeta, mzamora)@um.es.

Victor Chang is with the Xi'an Jiaotong Liverpool University, China, e-mail: ic.victor.chang@gmail.com

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2016.2605581, IEEE Transactions on Industrial Informatics

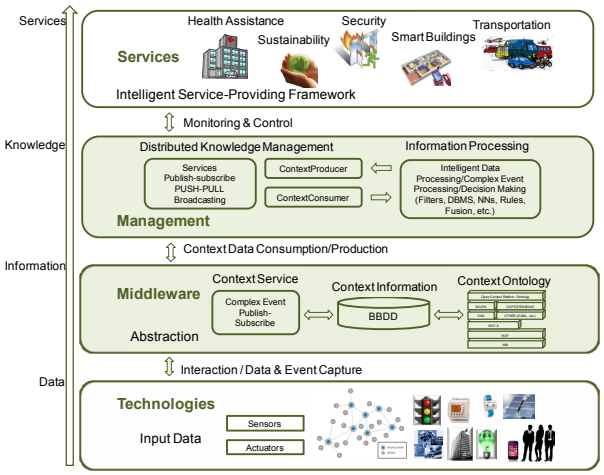


Figure 1: Layers of the base architecture for smart city services

II. IoT-based Architecture for Smart Cities

In this section we enumerate the main challenges that most current smart cities still have to face. Then, motivated by these challenges, we make a proposal of a general IoT-based architecture for smart city applications.

A. Challenges of Smart Cities

The global challenges that smart cities still have to face can be summarized in the following way:

- Sensors integration and abstraction capability. Provide means to integrate different sensor types in a common platform taking into account the different technologies, legacy systems and communication protocols with focus on IPv6 solutions.
- Individual intelligence and local reasoning. Apart from data fusion, more complex data processing can be implemented by smart objects.
- Learning and adaptation. Most of the patterns generated in smart cities are sensitive to contextual changes and are able to learn and adapt themselves to such changes as well as to human dynamicity.
- Dynamic human centric services. This work designs and implements smart mobility and smart building services that use the patterns generated to provide customized and efficient services taking into account the dynamicity of the citizens' behavior.
- User privacy and security control mechanisms. In the context of smart cities it is important to manage the way the user is able to control its data and how they are exposed to third parties and applications.

B. IoT-based Architecture

Several layers compound the proposed platform that was created with the goal of serving to many applications of smart cities. In Figure 1 is depicted this layered IoT-based architecture, which are detailed below.

1) Technologies Layer: In the basis part of Figure 1 it is observed that a plethora of sensors and network technologies provide the input attributes using wireless sensor networks, wired sensors, gateways, etc. which can be self-configured and remotely controlled through the Internet. Dealing with our first application that consists on the instantiation of the architecture for building management systems (BMS), in this layer it is gathered information from sensors and actuators deployed in strategic points of the building. But the aforementioned data sources in smart cities are not limited to static devices reporting measurements associated to a particular location, there are also moving ones capable to deliver measurements at different points within a geographical area. This is mainly due to the rapid development of wireless technology, mobile sensor networks and, above all, the advent of smartphones [5]. Although approaches based on mobile-phone sensing require a demanding usage of the communication, location and other attributes of the smartphone, which can bother some people due to battery draining [6], data captured by static, mobile and smart-phone sensors can be extended or enriched with the data generated by several social-media channels - like Twitter or Facebook - giving rise to a new generation of *soft sensors* from which extract relevant knowledge [7]. As a result, an alternative course of action aims at mining relevant knowledge from users on the basis of non-intrusive ways to obtain data, for example, transit cards in public transport scope.

2) Middleware Layer: The first layer provides us with a wide variety of data, so it is needed a second layer where all collected data from seamless sources are expressed in the same way, this is done in the middleware layer. The context information can be collected in an ontology defined according to the model that represents the knowledge of the specific application domain. Thus, for our energy efficiency semantic model, the devices and building concepts are borrowed by the SAREF ontology [8]. The agents representation is made using the DUL ontology [9], while the observation values of the monitored sensors are represented based on the SSN ontology [10]. However, when it comes to process the incoming data

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2016.2605581, IEEE Transactions on Industrial Informatics

in a real-time manner, it is necessary to use a lightweight representation. As a matter of fact, [11] describes sensor-data representation using a simple attribute-value schema for event-based systems.

3) Management Layer: After having extracted information from the previous layers, the management layer is in charge of determining decisions bearing in mind the target services provided in smart cities. Different big data analytic techniques can be used for the intelligent decision making processes. Algorithms like Artificial Neural Networks (ANNs) using backpropagation methods [12] and Support Vector Machines (SVMs) [13] are good to solve non-linear problems, making them very applicable to build energy prediction issues, ranging from those associated to lighting and heating, ventilation and air condition (HVAC) [14] to the prediction of the heating energy requirements [15]. For optimization problems in Building Management System (BMS) addressing energy efficiency, Genetic Algorithms (GAs) constitute a commonly applied heuristic that can be used in several optimization scenarios such as scheduling cooling operation decisions [16]. Regarding to the smart public transport application, the extraction of users behaviors from transition records have been studied by using different algorithms and techniques like maximum likelihood estimation [17], probabilistic models [18], conditional random fields [19], graphical information system (GIS)-based processing [20] or Database Management System (DBMS)-based processing [21].

4) Services Layer: Finally, the upper layer (Figure 1) shows some examples of smart city services that can be provided following the proposed architecture. Thus, this architecture can be applied to provide applications of smart cities like environmental monitoring, energy efficiency in buildings and public infrastructures [22], environmental monitoring [23], traffic information and public transport, locating citizens, manage emergencies, saving energy and other services. These actions can either involve citizens or be automatically set.

III. Smart Campus of the University of Murcia

The University of Murcia (UMU) has three main campus and several facilities deployed throughout different cities in the Region of Murcia. One of these campus is currently serving as pilot of two European Projects, the SMARTIE [24] and the ENTROPY [25] project. The goal of this use case of smart city is to provide a reference system able to manage intelligently the energy use of the most relevant contributor to the energy use at city level, i.e. buildings. The BMS implemented as part of this smart campus adapts the performance of automated devices through decisions made by the system and the interaction with occupants in order to keep comfort conditions while saving energy. We start by the most representative source of energy consumption at building level: HVAC systems.

A. System Overview

Using a BMS system, it is possible to predict users future behaviour from their recorded activities that are measured with sensors. This information allows us to provide convenient environments looking for keeping their comfort while saving energy. The first need for a building to become smart is to know location of occupants. Once solved the indoor localization problem, it is time to propose a solution to the energy efficiency of buildings associated to the thermal comfort provisioning service related to the HVAC management. For this, energy consumption models of the building need to be generated and used to implement the optimization mechanism able to maximize comfort at the same time that energy consumption is minimized. Therefore, the different problems addressed in this scenario of smart city through the application of big data techniques are:

1) Indoor localization estimation.
2) Building energy consumption prediction.
3) Comfort provisioning and energy saving through an optimization problem.

In the following subsections these problems are described with more details, as well as the techniques implemented and the results obtained.

B. Indoor Localization Estimation

As well as considering the information concerning to the identification and location of the building's occupants, it is necessary to reach the required accuracy in the location in order to provide the indoor services in a comfortable and energy efficient way. Our technological solution to cover the localization needs (i.e. those required by smart buildings to provide occupants with customized comfort services) is based on a single active RFID system and several Infra-Red (IR) transmitters. In Figure 2 we can observe the data exchange carried out among the different technological devices that compose our localization system.

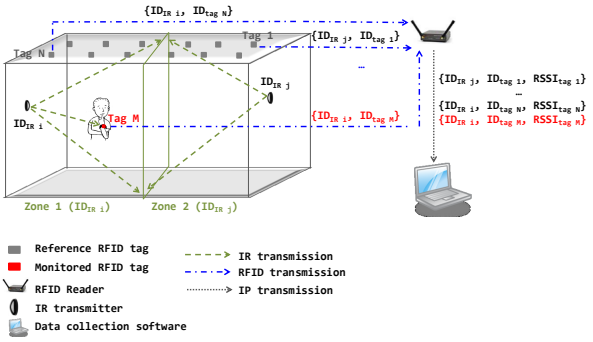


Figure 2: Localization scenario

The final mechanism implemented to solve the indoor localization problem is shown in Figure 3. In this figure, we can see that the first phase of the mechanism is the space division through the installation of IR devices in the walls of the building area where localization wants to be solved. Therefore, for each space division, there is an IR identifier value (ID_{ir}) associated to this region. For each one of these regions, we implement a regression method based on Radial Basis Functions (RBF) networks. The RBF estimates user positions given different RFID tags situated in the roof. Then,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2016.2605581, IEEE Transactions on Industrial Informatics

after the position estimation using the RBF network, a Particle Filter (PF) is applied as a monitoring technique, which takes into account previous user position data for estimating future states according to the current system model.

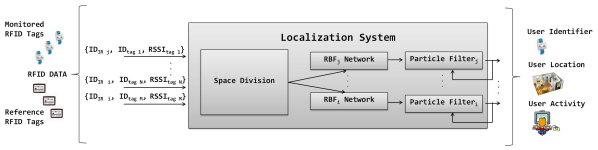


Figure 3: Data processing for location estimation

The PF used in this work is slightly different from its generic definition (which can be found in [26]). The main difference of our filter is in the correction stage. In this stage, the generic definition of the PF applies the resampling using the Sequential Importance Sampling (SIS) algorithm [26] to carry out the filtering of such particles which minimize the deviation of their predicted trajectory. In our implementation, in addition to apply the SIS algorithm to correct the particles positions, we also use in this step the information about the specific IR region at a given instant of time to benefit those particles which fall inside this area. Therefore, before applying the SSI algorithm, we filter according to the coverage area of the IR transmitter identified by the monitoring RFID tag. The main advantage of this constraint is the faster convergence of the filter, because extra information is available to carry out the correction stage of the filter.

C. Building Energy Consumption Prediction

The energy performance model of our BMS is based on the *CEN Standard EN 15251* [27]. This standard proposes the criteria of design for any BMS. It establishes and defines the main input parameters for estimating building energy requirements and evaluating the indoor environment conditions. The inputs considered to solve our problem are the data coming from the RFID cards of users, the user interaction with the building automation system through the control panels or the web access, environmental parameters coming from temperature, humidity and lighting sensors installed in outdoor and indoor spaces, the consumption energy sensed by the energy meters installed in the building, and the generated energy sensed by the energy meters installed in the solar panels deployed in our testbed. After collecting the data it is mandatory to continue with their cleaning, preprocessing, visualization and correlation calculation in order to find determining features, which can be used to generate optimal energy consumption models of buildings (management layer of the architecture presented in Section II). Over the input set, we perform the standardization and reduction of data dimensionality using Principal Components Analysis (PCA) [28], identifying the directions in which the observations of each parameter mostly vary.

Regarding the big data techniques that have been already applied successfully to generate energy consumption models of buildings in different scenarios (as such we mentioned in the management layer of the architecture presented in Section II-B3), we propose to evaluate the performance of Multilayer Perceptron (MLP), Bayesian Regularized Neural Network (BRNN) [29], SVM [30] and Gaussian Processes with RBF Kernel [31]. They were selected because of the good performance that all of them have already provided when they are applied to building modelling. All these regression techniques are implemented following a model-free approach, which is based on selecting - for a specific building - the optimal input set and technique, i.e. such input set and technique that provides the most accurate predictive results in a test dataset. In order to implement this free-model approach, we use the R [32] package named CARET [33] to train the energy consumption predictive algorithms, looking for the optimal configuration of their hyper-parameters (more information can be found in [34]). The selected metric to evaluate the models generated for each technique using test sets is the well-known RMSE (Root-Mean-Square Error), whose formulation appears in Eq. (1). This metric shows the error by means of the quantity of KWh that we deviate when predicting.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{1}$$

But in order to get a better understanding of the uncertainty of the model, we also show its coefficient of variation (CVRMSE). This coefficient is the RMSE divided by the mean of the output variable (energy consumption) for the test set (see Eq. (2)), giving us a percentage of error adjusted to the data, not just a number in general terms.

$$CVRMSE = \frac{RMSE}{\bar{y}} \tag{2}$$

D. Optimization Problem

Once the building energy consumption is modelled, we focus on the optimization of the HVAC operation trying to keep comfort conditions at the same time that energy consumption restrictions are considered. As starting point, we establish the comfort extremes considering location type, user activity and date [35]. Understanding the building thermal and energetic profiles allows us to quantify the effects of particular heating/cooling set point decisions. To derive a heating or cooling schedule, it is necessary to formulate the target outcome. In our buildings, it is possible to:

1) optimize the indoor temperature during occupation, i.e. minimize the building temperature deviations from a target temperature,
2) minimize daily energy consumption, or
3) optimize a weighted mixture of the criteria, a so-called multi-objective optimization problem.

The definition of building temperature deviation influences the results strongly: taking the minimum building temperature will result in higher set point choices and higher energy use than using, for instance, the average of indoor temperatures. Constraints on maximum acceptable deviation from target comfort levels or an energy budget can be taken into account to ensure required performance. In our optimization problem,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2016.2605581, IEEE Transactions on Industrial Informatics

we apply GA using the implementation provided by R (the "genalg" package [36]), to provide schedules for heating/cooling setpoints using the predictive building models (comfort and energy consumption models).

E. Evaluation and Results

1) Scenario of Experimentation: The reference building where our BMS for energy efficiency is deployed is the Technology Transfer Centre (TTC) of the UMU[1]. Every room of this building is automated through a Home Automation Module (HAM) unit. It permits us to consider a granularity at room level to carry out the experiments.

2) Evaluation. Indoor localization mechanism: Different tracking processes are carried out in the environments considered in our tests (the TTC building), applying for this the implementation of our PF. In Figure 4 an example of some tracking processes are carried out considering transition between different spaces of the TTC. For these paths, our system was configured to acquire data every $T = 10$ s. Taking into account the target location areas involved in comfort provisioning (lighting and thermal comfort, represented in different colors), and the real and estimated location data provided by our mechanism.

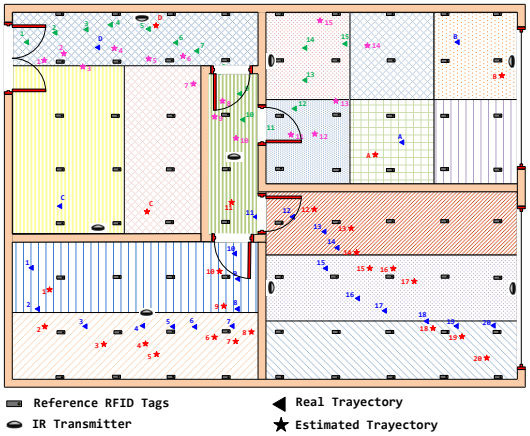
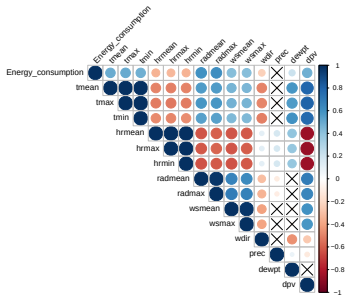


Figure 4: Tracking processes with a reference tag distribution of 1m x 1m

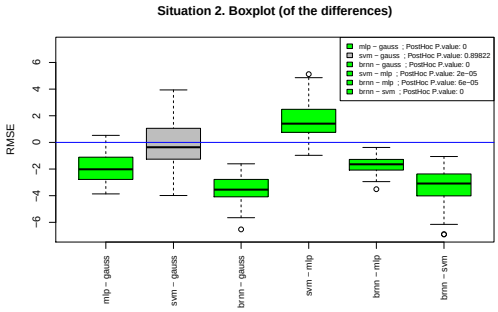
Thus, with a $1m$ x $1m$ distribution of reference RFID tags placed on the roof of the test room, a 65% success percentage in localization is obtained having an error lower than 1m. 98% of cases have as much 2.5m. of error. Therefore, it can be safely said that our localization system is able to track users with a sufficient level of accuracy and precision for the location requirements associated with the comfort and energy management in buildings. More details about this indoor localization system can be found in [37].

3) Evaluation. Energy consumption prediction: In Figure 5(a) it is shown the correlation heatmap between the electrical consumption of the TTC building and the outdoor environmental conditions. It is observed that energy consumption

[1] www.um.es/otri/?opc=cttfuentealamo



(a) Correlation heatmap between consumption and outdoor environmental conditions



(b) Boxplots comparing models pairwise (situation 2)

Figure 5: Modeling results

correlates significantly ($\alpha = 0.95$) and positively with temperature, radiation, wind speed variables, vapour pressure deficit and dew point; and negatively with wind direction and humidity variables. This means that we can use safely these variables as inputs of the energy consumption model of our reference building, because they have clear impact in the energy consumption. Otherwise, precipitation is so unusual that they don't have an association with the output.

Also, a logic differentiation between temporal situations has been considered in order to label behaviour. Situation 1: holidays and weekends; situation 2: regular mornings; and, situation 3: regular afternoons. The non-parametric Kruskall Wallis test shows that energy consumption differs significantly between situations ($H(2) = 547.7$, $p < 0.01$). Also, the post hoc pairwise comparisons corrected with Holm's method retrieve a p-value smaller than 0.01, supporting the decision of creating 3 different models [38]. Thus, for each of the three situations identified for the TTC building, we have evaluated not only the punctual value of RMSE, but also we have validated whether one learning algorithm out-performs statistically significantly the others using the non parametric Friedman test [39] with the corresponding post-hoc tests for comparison.

Let x_i^j be the i-th performance RMSE of the j-th algorithm, for this building, we have used 5-times 10-fold cross validation, so $i \in \{1, 2, ..., 50\}$ and four techniques, so $j \in \{1, 2, 3, 4\}$. For every situation, we find significant differences ($\alpha = 0.99$) between every pair of algorithms,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2016.2605581, IEEE Transactions on Industrial Informatics

except for SVM and Gauss RBF ($p > 0.01$), as it is shown in Figure 5(b) for the particular case of situation 2.

The three models have in common that BRNN yields a better result than the other tested techniques, based on the RMSE metric. Thus, BRNN is able to generate a model with a very low mean error of 25.17 KWh - which only represents the 7.55% of the sample (this is the most accurate result) in terms of the CVRMSE. And for the worst case, BRNN provides a mean error of 43.76 KWh - which represents the 10.29% of the sample in the reference TTC building - that is acceptable enough considering that our final aim is to save energy.

4) Evaluation. Optimization mechanism: To evaluate our GA-based optimization strategy, controlled experiments were carried out in the TTC building with different occupant's behaviours. The results show that we can accomplish energy savings between the 15% and 31%. Trying to validate the applicability of our proposal, we have also made experiments in a different scenario with limited monitoring and automation technologies, achieving energy saving of about the 23%.

IV. Public Tram Service of Murcia City

The second scenario is focused on the information analysis related to use of the tram service of the Region of Murcia [40]. In this case, the main goal was to perform a profiling process of the trips carried out by the users of such public service. For that aim, a fuzzy clustering algorithm is used to automatically extract tram user's profiles. Bearing in mind the architecture introduced in Section II, this system is enclosed in the management layer. The main tasks needed to reach the goal are explained in the following subsections.

A. Generation of the trip data set

According to the tram experts, information relevant to trip profiling must include data about: time (in terms of day of the week and time of the day), origin and destination stations and approximate age of the traveller. This information is being continuously recorded in different databases of the tram service. Nevertheless, certain operations of joining, transformation and preprocessing (discretization and numerization) have been performed in order to compile all this information into a set of tuples susceptible of feeding the subsequent fuzzy clustering algorithm. The two most remarkable operations are the following:

On the one hand, according to the infrastructure of the tram service, users only need to swipe the smart card when they get into the tram. Hence, the recorded data only comprises transactions at the origin of each user's trip so it can be regarded as incomplete. In order to deal with this incompleteness, a well known solution is the **trip-chaining method** which focus on recovering the origin and destination of the trips. In this case, such a method is based on the assumption that a traveller who takes the tram at an origin station, OS, ended their previous trip on that station OS. Due to the event-based nature of the card records, the Complex Event Processing (CEP) paradigm [11] was adopted to come up with a palette of event-condition-action rules to uncover the trips. While the condition part of the rules performs a match between consecutive records of the same traveller following the aforementioned trip-chaining method, the action part generates a new trip tuple (comprising the origin and destination stations) in case the condition is fulfilled.

On the other hand, as clustering techniques are based on distance calculations among data, a set of numbered (and ordered) geographical areas, each one covering some close stations are identified by the tram experts. Then, instead of having nominal values for origin and destination features these numbered areas make it easier to calculate the distance about tuples in the clustering process.

In summary, the tuples composing the data set for the subsequent clustering task are composed by the following attributes: tt_e*:{travellerAge, dayOfTheWeek, hourOfTheDay, originArea, destArea}*

B. Trip profiling

Clustering mechanisms are suitable when it comes to find out the most representative trips profiles. For that aim, the Gustafson-Kessel Clustering Method (GKCM) has been chosen since it is able to identify arbitrarily oriented ellipsoidal fuzzy clusters unlike, for instance, the Fuzzy C Means clustering Method, which impose spherical shapes to the data clusters. After the clustering task the identified prototypes (centroids) will summarize the whole data set of trips. GKCM requires to be supplied with the quantity of potential clusters (c). This is an important parameter since it determines the ability of the potential centroids to represent the real underlying structure of the data.

Therefore, several GKCM executions were performed with different values of c and the *goodness* of the different identified set of clusters was measured. One of the most used measurement is the one proposed in [41] and denoted here as r_{cs}. This magnitude quantifying both the total compactness within clusters and the total separation among them being the greater the better.

Once the number of clusters c has been decided on the basis of r_{cs}, GKCM is executed in order to find the c profiles that best represent the trip data set. Nevertheless, when exceed a time tp_{max} or a number of trips nt_{max} the algorithm is recomputed in order to detect new profiles which could rise up.

C. Evaluation and Results

The subject of evaluation is the tram service of the region of Murcia (Spain), which includes 18-km railway and 28 stations (see Figure 6). Figure 7 depicts the set of predefined geographical areas used in the experiment.

The evaluated dataset contained 378719 trips from 23400 users in November, 2013. For our experiment, the system was able to uncover 110697 trips. Expert knowledge was used to define the types of days and times of the day used in the aforementioned data pre-processing step as [*Monday-Thursday, Friday, Saturday, Sunday*] and [*0-6, 6-10, 10-12, 12-16, 16-20, 20-00*]. As a result, a generated TT_e dataset was split up into 4 different subsets based on the fact that traveller profiles depend on the day of the week (regarding, for example, differences of traffic flow between regular workdays

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2016.2605581, IEEE Transactions on Industrial Informatics

Algorithm 1: Cluster-based Trip profiling process.

Input: TT: dataset of raw trip tuples.
Output: P_{TT}: Traveller profiles extracted from TT.

```
if t_now − t_prev > tp_max ∨ | TT | − | TT_prev |> nt_max then
    TT_e ← preProcessing(TT)
    foreach c ∈ {2, .., c_max} do
        clust_c = GKCM(TT_e, c)
        if clust_c.r_cs < r_cs^min then
            r_cs^min ← clust_c.r_cs
            P_TT ← clust_c.centroids
    t_prev ← t_now
    TT_prev ← TT
    return P_TT
```

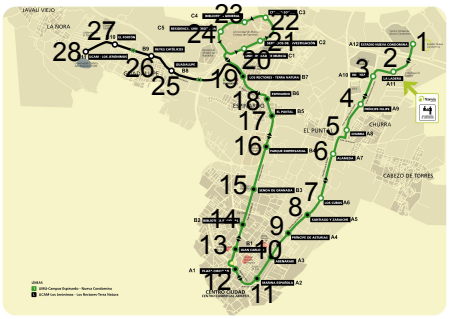


Figure 6: Line map of the tram service in Murcia.

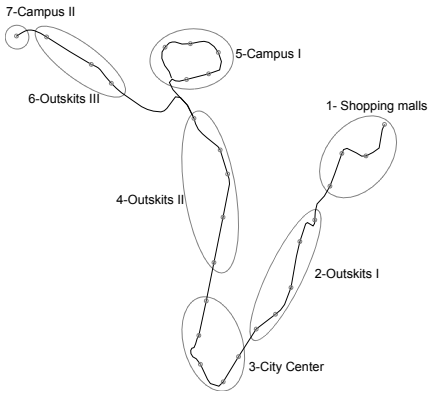


Figure 7: Geographical regions for the numerization of tuples' station fields.

and weekends). Next, the GKCM was launched with each of these subsets with different number of clusters.

In Figure 8, the cluster validation ratio r_{cs} is shown for every TT_e subset, being the lower value the better. As it can be observed, while the optimal cluster partition is reached at $c = 5$ for the Monday-Thursday subset, for the remaining subsets minima r_{cs} values are reached at higher number of clusters c. In other words, a higher number of profiles is needed to represent the weekend trips. This is reasonable given that most

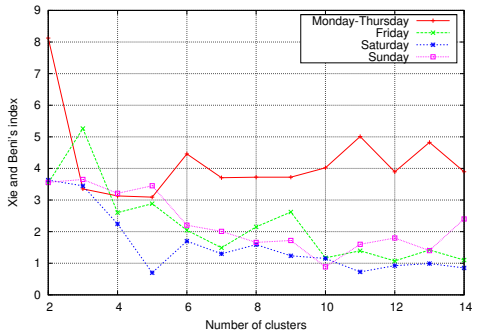


Figure 8: Cluster-validation rate for different cluster partitions.

people postpone leisure activities to the weekend and given that there exist a quite variety of leisure activities that can be done at different hours of the day.

As Table 1 shows, GKCM extracts five profiles for Monday-Thursday trips. Profiles 1 and 2 correspond to young people travelling in the morning to go towards one of the university zones from the station close the inner city. Besides, profile 5 represents a kind on traveller going back home from the university from 4 to 8 PM. Finally, profiles 3 and 4 correspond to middle-young age people (28-33 years) that take the tram around the outskirts and city center environments. These could reflect people going from residential areas.

Lastly, the heatmap shown in Figure 9 represents the membership of the Monday-Thursday trips to the defined profiles. If we interpret this plot as a time-framed sequence, a great amount of the traffic focuses on the right side of the line, which connects the city center and the university areas. Nevertheless, such load is more spread along the whole line during the evening.

V. Discussion

In this paper we propose a general IoT-based architecture which can be implemented for different applications of smart cities. This architecture is modeled in four layers, being the third one - the management layer - the layer where big data techniques are implemented to provide the different services offered then in the corresponding service layer (last layer).

The big data paradigm can be understood through the lens of 7 V's [42] (challenges). Regarding the application of different big data techniques to the specific scenarios of smart cities presented in this paper, we have overcame the challenge of *velocity* by collecting data hourly in the smart building application (consumption of energy, outdoor environmental conditions) and even in sorter intervals of time for the public transport application (many people validates their transit cards within seconds). Although we haven't tackled *volatility*, it is clearly a goal when looking for the real-time smart city because behavioural scenarios like ours change depending on many social aspects. The *veracity* of the data is guaranteed by the exhaustive pre-processing steps included in the modeling process. We have extracted *value*, making sense of the wide mentioned *variety* of data, and with the described analysis

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2016.2605581, IEEE Transactions on Industrial Informatics

Profile	Age	Origin Area	Dest. Area	Time of the day
P1	23.37	*City Center*	*Campus I*	0-6
P2	25.74	*City Center*	*Campus I*	6-10
P3	28.22	*Outskits II*	*City Center*	12-16
P4	32.77	*Outskits I*	*Outskits II*	6-10
P5	22.20	*Campus I*	*City Center*	16-20

Table 1: Monday-Thursday trips' profiles.

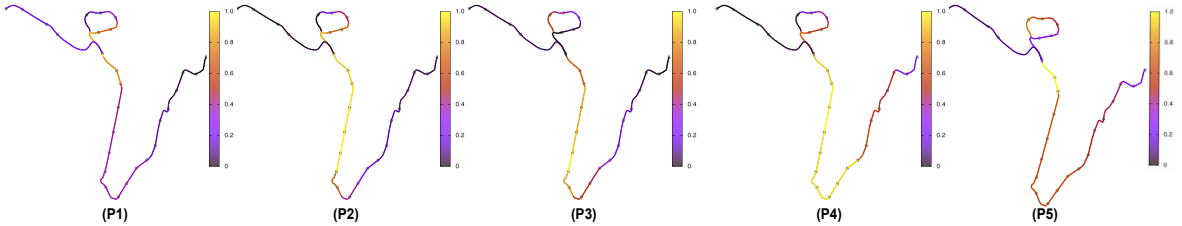


Figure 9: Tram-line heat-map of the five profiles for Monday-Thursday trips.

Smart City Application	Data	Information	Knowledge	Services
Smart Campus	IR Sensors. RFID tags. Environmental Sensors. Weather Station. Presence Sensors. Energy Consumption Meters. Weather Forecast	Data Transformation through SAREF ontology [8], DUL ontology [9] and SSN ontology [10]	Data Modelling. Predictive Regression (RBFs, SVM, ANN, RF, ARIMA). Tracking algorithm (PFs). Optimization Mechanism (GA)	Indoor localization. Building energy consumption prediction. Energy saving through the HVAC operation optimization
Public Tram Service	Mobile Sensors. Smart Cards	CEP-based filtering. Event Processing in Action [11]	Fuzzy Clustering	Infrastructure monitoring. Mobility patterns.

Table 2: Main features of the two architecture instantiations

and techniques, we have *validated* their usability for solving different problems of smart cities with high accuracy.

In both applications tackled in this paper, the huge *volume* of historical data is being stored using a NoSQL data base. At the moment, the storage system is been adapted so as to be compliant with the FI-WARE architecture[2], that intends to ease the development of novel applications based on the Future Internet. In particular, the Orion Context Broker (OCB)[3] and the COMET[4] modules are used in order to store in a NoSQL repository the historical data comprising the measurements from the different data sources.

On the whole, both instantiations of the architecture described above are summarized in Table 2. In the next subsections we summarize the main benefits obtained after applying the most suitable big data techniques to the two scenarios of smart cities addressed in this work.

A. Benefits of Big Data Applications in Smart Buildings for Energy Efficiency

Here we summarize the main findings extracted from all the experiments and analysis carried out during the application of big data techniques to the smart campus of the UMU.

[2]https://www.fiware.org [Available Feb. 2016]

[3]http://catalogue.fiware.org/enablers/publishsubscribe-context-broker-orion-context-broker [Available Feb. 2016]

[4]https://github.com/telefonicaid/fiware-sth-comet. [Available Feb. 2016]

1) **The resolution of the indoor localization problem.** Applying regression techniques based on RBFs and a tracking algorithm applying PFs to data coming from RFID and IR sensors installed in buildings, it was possible to solve the indoor localization problem with a mean accuracy of 1.5 m. Then, indoor localization data can be used to provide customized services in buildings.
2) **The resolution of the building energy consumption estimation.** Applying PCA and BRNN techniques to data related to outdoor environmental conditions and energy consumption of buildings, it was possible to generate energy consumption predictive models of buildings with a very low mean error of 43.76 KWh - which only represents the 10.29 % of the sample - in the worst case. Then, energy consumption predictions can be used to design the optimal strategies to save energy in buildings.
3) **The resolution of the optimization problem related to the maximization of thermal comfort and minimization of energy consumption in buildings.** Applying optimization methods based on GAs to optimize the energy consumption of buildings meanwhile comfort conditions are satisfied, and after including user localization data and user comfort preference prediction, it was possible to get energy savings in heating of about 23% compared with the energy consumption in a previous month without any energy BMS.

B. Benefits of Big Data Applications in Urban Pattern Recognition to Improve Public Tram Service

After applying Big Data techniques to the urban pattern extraction in the public tram service, all the results from the experiments allowed the service staff to draw up quite interesting conclusions. These are summarized below:

1) **Regarding the resolution of the trip extraction**. The formal discovery of the stations' load in terms of trips' origin and destination would allow the service provider and the city council to better plan the whole public transport service of the city. This way, the more important stations might be considered as "hub" points where commuters can easily transfer from tram to another kinds of transport. Moreover, such an information could be also useful so as to forecast future infrastructure needs in each part of the tram line (e.g. location and number of places of new parking lots for bicycles close to tram stations).
2) **Concerning the resolution of the urban profiles generation**. Experiments pointed out the importance of undergraduates as tram users. Hence, most of the traffic load concentrated in the line between the city center and the campuses. This was really helpful in order to design promotional campaigns for these type of travellers. Moreover, results also confirmed that the line segment towards the shopping-mall areas was underused. Thus, campaigns to promote the use of the tram to go shopping was also considered.

VI. Conclusions and Future Work

This paper displays the benefits of applying big data techniques over data originated by IoT-based devices deployed in smart cities. A general architecture modelled in four layers is proposed to be applied in smart city applications considering big data issues. As part of this overview, a differentiation between static and mobile data sources is made, proposing for each one of them suitable techniques to extract relevant knowledge from their data. Then, we describe two big data applications for smart city services. Specifically, the services of energy efficiency and comfort management in the buildings of a smart campus, and the public transport service of a city. In the first scenario of smart city we have demonstrated that, after applying appropriate big data techniques to problems like indoor localization, energy consumption modeling and optimization, we are able to provide mean energy savings of 23% per month, while indoor comfort is ensured. Regarding to the urban pattern recognition carried out using data related to the public tram service of the city of Murcia, experiments were performed to confirm that the proposed patterns ended up being of great interest for the service provider in order to better understand how travellers make use of the transportation system. This was fairly useful in order to come up with better planning protocols and more tempting promotional campaigns.

The ongoing work is focused on the inclusion of people behaviour during the operational loop of this kind of systems for smart cities. Thus, for the case of smart building applications, users will be encouraged to participate in an active way through their engagement to save energy. On the other hand, in the case of the public tram service, data coming from crowdsensing initiatives will be integrated to improve the estimation of the urban mobility patterns.

Acknowledgments

This work has been funded by the Spanish Seneca Foundation by means of the PD program (grant 19782/PD/15), the MINECO TIN2014-52099-R project (grant BES-2015-071956) and ERDF funds, and by the European Commission through the H2020-ENTROPY-649849 and FP7-SMARTIE-609062 EU Projects.

References

[1] N. Komninos, "Intelligent cities: variable geometries of spatial intelligence," *Intelligent Buildings International*, vol. 3, no. 3, pp. 172–188, 2011.
[2] L. Atzori, A. Iera, and G. Morabito, "The internet of things: A survey," *Computer networks*, vol. 54, no. 15, pp. 2787–2805, 2010.
[3] L. Da Xu, W. He, and S. Li, "Internet of things in industries: a survey," *Industrial Informatics, IEEE Transactions on*, vol. 10, no. 4, pp. 2233–2243, 2014.
[4] R. Iqbal, F. Doctor, B. More, S. Mahmud, and U. Yousuf, "Big data analytics: Computational intelligence techniques and application areas," *Int. J. Inf. Manage*, pp. 10–15, 2016.
[5] Z. Yan and D. Chakraborty, "Semantics in mobile sensing," *Synthesis Lectures on the Semantic Web: Theory and Technology*, vol. 4, no. 1, pp. 1–143, 2014.
[6] A. Carroll and G. Heiser, "An analysis of power consumption in a smartphone." in *USENIX annual technical conference*, 2010, pp. 1–14.
[7] M. Kantardzic, *Data mining: concepts, models, methods, and algorithms*. John Wiley & Sons, 2011.
[8] L. Daniele, F. den Hartog, and J. Roes, "Created in close interaction with the industry: The smart appliances reference (saref) ontology," in *Formal Ontologies Meet Industry*. Springer, 2015, pp. 100–112.
[9] K. Janowicz and M. Compton, "The stimulus-sensor-observation ontology design pattern and its integration into the semantic sensor network ontology," in *Proceedings of the 3rd International Conference on Semantic Sensor Networks-Volume 668*. CEUR-WS. org, 2010, pp. 64–78.
[10] M. Compton, P. Barnaghi, L. Bermudez, R. GarcíA-Castro, O. Corcho, S. Cox, J. Graybeal, M. Hauswirth, C. Henson, A. Herzog *et al.*, "The ssn ontology of the w3c semantic sensor network incubator group," *Web Semantics: Science, Services and Agents on the World Wide Web*, vol. 17, pp. 25–32, 2012.
[11] O. Etzion and P. Niblett, *Event Processing in Action*, 1st ed. Greenwich, CT, USA: Manning Publications Co., 2010.
[12] T. Maniak, C. Jayne, R. Iqbal, and F. Doctor, "Automated intelligent system for sound signalling device quality assurance," *Information Sciences*, vol. 294, pp. 600–611, 2015.
[13] A. H. Neto and F. A. S. Fiorelli, "Comparison between detailed model simulation and artificial neural network for forecasting building energy consumption," *Energy and Buildings*, vol. 40, no. 12, pp. 2169–2176, 2008.
[14] H.-x. Zhao and F. Magoulès, "A review on the prediction of building energy consumption," *Renewable and Sustainable Energy Reviews*, vol. 16, no. 6, pp. 3586–3592, 2012.
[15] B. B. Ekici and U. T. Aksoy, "Prediction of building energy consumption by using artificial neural networks," *Advances in Engineering Software*, vol. 40, no. 5, pp. 356–362, 2009.
[16] F. Ascione, N. Bianco, C. De Stasio, G. M. Mauro, and G. P. Vanoli, "Simulation-based model predictive control by the multi-objective optimization of building energy performance and thermal comfort," *Energy and Buildings*, vol. 111, pp. 131–144, 2016.
[17] W. Wang, J. P. Attanucci, and N. H. Wilson, "Bus passenger origin-destination estimation and related analyses using automated data collection systems," *Journal of Public Transportation*, vol. 14, no. 4, p. 7, 2011.
[18] M.-P. Pelletier, M. Trépanier, and C. Morency, "Smart card data use in public transit: A literature review," *Transportation Research Part C: Emerging Technologies*, vol. 19, no. 4, pp. 557–568, 2011.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TII.2016.2605581, IEEE Transactions on Industrial Informatics

[19] N. Yuan, Y. Wang, F. Zhang, X. Xie, and G. Sun, "Reconstructing Individual Mobility from Smart Card Transactions: A Space Alignment Approach," in *2013 IEEE 13th International Conference on Data Mining (ICDM)*, Dec. 2013, pp. 877–886.
[20] M. A. Munizaga and C. Palma, "Estimation of a disaggregate multimodal public transport origin–destination matrix from passive smartcard data from santiago, chile," *Transportation Research Part C: Emerging Technologies*, vol. 24, pp. 9–18, 2012.
[21] J. P. Attanucci and N. H. Wilson, "Bus passenger origin-destination estimation and related analyses using automated data collection systems," *Journal of Public Transportation*, vol. 14, no. 4, p. 131, 2011.
[22] P. Palensky and D. Dietrich, "Demand side management: Demand response, intelligent energy systems, and smart loads," *Industrial Informatics, IEEE Transactions on*, vol. 7, no. 3, pp. 381–388, 2011.
[23] S. Fang, L. Da Xu, Y. Zhu, J. Ahati, H. Pei, J. Yan, and Z. Liu, "An integrated system for regional environmental monitoring and management based on internet of things," *Industrial Informatics, IEEE Transactions on*, vol. 10, no. 2, pp. 1596–1605, 2014.
[24] EU Smartie Consortium. (2013-2016) EU Smartie Project. [Online]. Available: http://www.smartie-project.eu
[25] EU Entropy Consortium. (2015-2018) EU Entropy Project. [Online]. Available: http://entropy-project.eu/
[26] A. Haug, "A tutorial on Bayesian estimation and tracking techniques applicable to nonlinear and non-Gaussian processes," *MITRE Corporation, McLean*, 2005.
[27] E. Standard *et al.*, "Indoor environmental input parameters for design and assessment of energy performance of buildings addressing indoor air quality, thermal environment, lighting and acoustics," *EN Standard*, vol. 15251, 2007.
[28] H. Abdi and L. J. Williams, "Principal component analysis," *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 2, no. 4, pp. 433–459, 2010.
[29] L. Hawarah, S. Ploix, and M. Jacomino, "User behavior prediction in energy consumption in housing using bayesian networks," in *Artificial Intelligence and Soft Computing*. Springer, 2010, pp. 372–379.
[30] Y. Fu, Z. Li, H. Zhang, and P. Xu, "Using support vector machine to predict next day electricity load of public buildings with sub-metering devices," *Procedia Engineering*, vol. 121, pp. 1016–1022, 2015.
[31] M. Alamaniotis, D. Bargiotas, and L. H. Tsoukalas, "Towards smart energy systems: application of kernel machine regression for medium term electricity load forecasting," *SpringerPlus*, vol. 5, no. 1, pp. 1–15, 2016.
[32] R Core Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, 2015. [Online]. Available: http://www.R-project.org/
[33] M. Kuhn, "Building predictive models in R using the caret package," *Journal of Statistical Software*, vol. 28, no. 5, pp. 1–26, 2008.
[34] A. González-Vidal, V. Moreno-Cano, F. Terroso-Sáenz, and A. F. Skarmeta, "Towards energy efficiency smart buildings models based on intelligent data analytics," *Procedia Computer Science*, vol. 83, pp. 994–999, 2016.
[35] J. A. Orosa, "A new modelling methodology to control hvac systems," *Expert Systems with Applications*, vol. 38, no. 4, pp. 4505–4513, 2011.
[36] E. Willighagen, "Genalg: R based genetic algorithm," *R package version 0.1*, vol. 1, 2005.
[37] M. V. Moreno, M. Zamora-Izquierdo, J. Santa, and A. F. Skarmeta, "An indoor localization system based on artificial neural networks and particle filters applied to intelligent buildings," *Neurocomputing*, vol. 122, pp. 116–125, 2013.
[38] J. M. Andy Field and Z. F. Niblett, *Discovering Statistics Using R*, 1st ed. Sage Publications Ltd, 2012.
[39] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *The Journal of Machine Learning Research*, vol. 7, pp. 1–30, 2006.
[40] [Online]. Available: www.tranviademurcia.es
[41] S. Miyamoto, H. Ichihashi, and K. Honda, "Algorithms for fuzzy clustering," *Methods in c-Means Clustering with Applications. Kacprzyk J, editor Berlin: Springer-Verlag*, 2008.
[42] M. Ali-ud-din Khan, M. F. Uddin, and N. Gupta, "Seven v's of big data understanding big data to extract value," in *American Society for Engineering Education (ASEE Zone 1), 2014 Zone 1 Conference of the*. IEEE, 2014, pp. 1–5.

Dr. M. Victoria Moreno received the B.S. (Hons.) and the M.S. degrees in Telecommunications Engineering in 2006 and 2009, respectively, both of them from the School of Telecommunication Engineering of Cartagena, Spain; and the Ph.D degree in Computer Science in 2014 from the University of Murcia, Spain. Currently, she is a post-doctoral researcher of the Seneca Foundation. Research interests include data analysis and modelling, and energy efficiency in smart environments.

Dr. Fernando Terroso-Sáenz graduated from University of Murcia with a degree in Computer Science in 2006. He also received the master degree in Computer Science at the same university in 2010. Then, he finished his PhD at the Dept. of Communications and Information Engineering in 2013. Since 2009, he have been working as a researcher in this group. His research interests include Complex Event Processing (CEP), Ubiquitous Computing and Intelligent Transportation Systems (ITSs).

Aurora González-Vidal graduated in Mathematics at the University of Murcia in 2014. In 2015 she got a fellowship to work in the Statistical Division of the Research Support Services, where she specialized in Statistics and Data Analyisis. During 2015, she started her PhD studies in Computer Science, focusing her research in Data Analytics for Energy Efficiency.

Dr. Mercedes Valdés-Vela received the Computer Engineering degree (1998) and the Ph.D. degree (Hons.) in Computer Science (2003) at the University of Murcia (Spain). In 2000, she started to work as research staff in the Dept. of Information Engineering and Communications. She is currently a full time assistant professor of the same Department. Her main research areas are Soft Computing, Complex Event Processing and Ambient Intelligence.

Dr. Antonio F. Skarmeta received the M.S. degree in Computer Science from the University of Granada, Spain, and the B.S. (Hons.) and the PhD degrees in Computer Science from the University of Murcia. Currently, he is a Full Professor in Dept. of Information and Communications Engineering at the same university. He is involved in numerous projects, both European and National. Research interests include mobile communications, artificial intelligence and home automation.

Dr. Miguel A. Zamora received the M.S. degree in Automation and Electronics and the Ph.D. degree in Computer Science in 1997 and 2003, respectively, both of them from University of Murcia, Spain. Currently, he is a Senior Professor in the Dept. of Information and Communication Engineering of the same university. His research interests include consumer electronics, home and building automation and sensor fusion.

Dr. Victor Chang is an Associate Professor (Reader) at Xi'an Jiaotong Liverpool University, China since June 2016. Within four years, he completed PhD (CS, Southampton) and PGCert (Higher Education, Fellow) part-time. He helps organizations in achieving good Cloud design, deployment and services. He won a European Award on Cloud Migration in 2011, best papers in 2012 and 2015, and numerous awards since 2012. He is one of the most active practitioners and researchers in Cloud Computing, Big Data and IoT in UK.

4.3 An open IoT platform for the management and analysis of energy data

ARTICLE IN PRESS

Future Generation Computer Systems ▮ (▮▮▮▮) ▮▮▮–▮▮▮

Contents lists available at ScienceDirect

Future Generation Computer Systems

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

An open IoT platform for the management and analysis of energy data

Fernando Terroso-Saenz *, Aurora González-Vidal, Alfonso P. Ramallo-González, Antonio F. Skarmeta

Department of Information and Communications Engineering, Computer Science Faculty, University of Murcia, Spain

HIGHLIGHTS

- IoT platform for the management of energy data in buildings.
- Includes several inner features to support data analytics in the energy domain.
- Based on the open IoT initiative FIWARE.
- Evaluated in a real pilot with comprising several buildings.

ARTICLE INFO

Article history:
Received 15 March 2017
Received in revised form 24 May 2017
Accepted 23 August 2017
Available online xxxx

Keywords:
IoT platform
Energy consumption
FIWARE
Data mining

ABSTRACT

Buildings are key players when looking at end-use energy demand. It is for this reason that during the last few years, the Internet of Things (IoT) has been considered as a tool that could bring great opportunities for energy reduction via the accurate monitoring and control of a large variety of energy-related agents in buildings. However, there is a lack of IoT platforms specifically oriented towards the proper processing, management and analysis of such large and diverse data. In this context, we put forward in this paper the IoT Energy Platform (IoTEP) which attempts to provide the first holistic solution for the management of IoT energy data. The platform we show here (that has been based on FIWARE) is suitable to include several functionalities and features that are key when dealing with energy quality insurance and support for data analytics. As part of this work, we have tested the platform IoTEP with a real use case that includes data and information from three buildings totalizing hundreds of sensors. The platform has exceed expectations proving robust, plastic and versatile for the application at hand.

1. Introduction

Several reports claim that residential and commercial buildings represent around 30%–40% of the overall energy consumption in Europe and in the United States [1,2]. Because of this, buildings are known to be the largest end-use energy contributor followed by transport and industry, and therefore they are a clear target for potentially reducing global energy consumption substantially.

Despite being great consumers, there is some evidence that shows that public and private buildings have not fully exploited all opportunities available to increase their energy efficiency. On the contrary, they suffer from a rather substantial energy waste that is partly due to inefficient heating, cooling, lighting and other power system (equipment) [3], due to bad use of the systems (behavior) [4] and due to poor fabric efficiency [5]. Although the implementations of measurements to improve the first or the third category can be rather expensive, it has been seen that soft measurements that focus on the change of behavior of buildings' users are cheap, but yet, can contribute greatly to the reduction of energy use [6].

In order to address the aforementioned inefficiencies due to lack of understanding on how the systems should be operated and other behavioral related aspects in the building sector, one could consider the use of Information and Communication Technologies (ICT) and, more specifically, of the Internet of Things (IoT). This new paradigm that also exists at the domestic level could be used as an instrument to make a realization of the so called *Smart Building*. In fact, it is foreseen that from 2 to 3 houses out of 10 will be equipped with up to 500 smart devices in the near future [7].

The installation of smart meters and In Home Energy Displays to make households aware of their energy consumption is not new [8,9]. The adoption of these devices seems to be an opportunity to exploit them for the reduction of energy use when looking at the available scientific literature (will be detailed later). However, one may also think that the technological effort to deploy such systems may be substantial and become a barrier to achieve this level of technification of the buildings. Nevertheless this technification seems to be happening naturally.

* Corresponding author.
E-mail address: fterroso@um.es (F. Terroso-Saenz).

http://dx.doi.org/10.1016/j.future.2017.08.046
0167-739X/

The large amounts of IoT data that will be coming from buildings in the near expected future will have to be analyzed to reveal insights that could help to obtain, expose and understand knowledge from buildings. In turn, this derived knowledge should be able to help to achieve meaningful energy saving strategies and interventions in the targeted buildings [10].

These wealth of information about energy use, offers a great opportunity according to some literature on energy feedback that suggests that intelligent feedback, (that with an extra larger of computation over simple observation) is an effective technique for the reduction of energy demands via behavioral change [11]. Only with a platform capable of making this possible, the implementation of this new paradigm will be successful.

In the IoT ecosystem, several platforms have emerged providing support from the sensorization stage to the stage of management and storage of the data in different forms [12]. In that sense, one of the most large-scale affords is the FIWARE platform, a key initiative of the Future Internet Public–Private Partnership (PPP) to create a well-aligned set of open enablers to receive, process, contextualize and publish IoT data from and for smart cities including from city-wide information to dwelling specific data.[1]

Despite all the reasons exposed before, little efforts have been made so far in order to adapt such platforms to building energy management. This energy ecosystem comprises a set of particularities that should be targeted in a specific manner. After analyzing the few examples of studies that have tried to tackle this problem, one can see that it exists a pressing need to apply different data mining techniques in the building energy domain mainly focusing on consumption prediction and pattern discovery or failure tolerance [13]. Thus, IoT energy platforms should include functions for data analysis among their features.

Although giving insighting knowledge behind data is an instrumental aspect of the wealth produced by the IoT, existing platforms are still limited when it comes to integrate data processing and analytic techniques suitable for IoT ecosystems [14]. This is a fundamental limitation of the state of the art as it is key to ensure that the platform will work on the new paradigm of providing tailored, real-time energy feedback to people. This also includes features to support the easy extension of platforms to allocate new data mining techniques comprising common steps in the data mining process. Examples of such features are built-in data-cleaning mechanisms for data pre-processing and storage solutions that would facilitate the execution of online and offline data mining algorithms.

All the aforementioned limitations have motivated us to envision, design, develop and validate what we called the IoT Energy Platform (IoTEP). The key strength of IoTEP is that it is, to our knowledge, the first holistic solution to large scale building energy data management from IoT.

Unlike existing IoT platforms, IoTEP is mainly oriented to support and ease the analysis of large amounts of heterogeneous energy data. A simplified overview of the platform IoTEP is shown in Fig. 1 representing its key features.

To begin with, IoTEP has been designed to easily retrieve either the most up-to-date readings of each sensor within a building, or to retrieve the historic data from such sensors. By means of these two types of access, the platform facilitates the application of both online and offline data analyses over the collected data. As we will see on further sections, this functionality is implemented with two FIWARE storage components, the ORION context broker and COMET. For both enablers, a NGSI-based information model has been defined in order to homogenize all the measured energy-related data.

[1] https://catalogue.fiware.org/.

Secondly, a real-time data cleaning module has been designed as a built-in component of IoTEP. With this, sensor readings are filtered by discarding potential outliers before injecting them in the storage components. This ensures a more efficient use of the resources. For this feature, we have followed a Complex Event Processing (CEP) approach that allows the real-time processing of event streams.

In addition to the above mentioned features, the platform includes also a mechanism to detect volatility changes in the incoming energy data. This mechanism intends to perceive meaningful shifts in such data that might need to re-launch the data-mining services that run within the platform.

Finally, IoTEP features a novel mechanism to automatically identify high-level areas in a building with certain energy-related similarities by means of clustering techniques. The benefit of these virtual areas is twofold. Firstly, they provide alternative representations of the energy status of a building beyond its physical structure; and secondly, they can help in the performance of other data mining analyses by reducing redundancies and defining different granularity levels in the captured sensor data.

Summarizing, the platform presented in this paper intends to be the first stage towards the full adaptation of the IoT paradigm in the retrieval, management and, above all, analysis of energy data in buildings. Considering the need of developing tools that are able to provide personalized real time feedback to change behaviors, and with them, have the potential to reduce energy use, IoTEP is intended to become the stepping stone for the development of such tools.

The paper is structured as it follows: Section 2 provides an overview of the state of the art in this research area. Section 3 looks into the IoT energy platform, including its architecture and its functional modules. Section 4 provides an evaluation of some of the features of the platform; and Section 5 concludes the paper with some final remarks and conclusions.

2. Related work

The present work is based upon two different lines of research, the management of energy data and the implementation of IoT platforms. Consequently, an overview of both lines is put forward in this section.

2.1. Energy data management systems

During the last years, some initiatives within the cloud computing domain have been made to intelligently manage energy data of buildings. In that sense, Zhou et al. [15] described a model for big-data energy management ranging from the collection and preprocessing of data to its further analysis and the final exposition to services. However, it only provides a theoretical approach.

From a practical perspective, the Dynamic Demand Response (D^2R) platform [16] makes use of public and private clouds combined with infrastructure and platform as a service for data storage. This platform was extended with *Cryptonite*, a repository to store sensitive *Smart Grid* data [17]. Then, different classes of data-driven forecasting models were generated on top of the whole platform with the purpose of carrying out energy prediction among others.

ElasticStream also provides a prototype solution for energy data management and analysis. In this case, the proposed mechanism transfers energy data to a cloud platform for further analysis on the basis of rate changes in the input data streams [18]. Moreover, Vastardis et al. [19] described a centralized architecture to monitor energy consumption in houses including features of pattern-matching related to the behavioral habits of the target users.

In the work of Ozadowicz [20], the authors propose different approaches to calculate the power demand related to energy

Please cite this article in press as: F. Terroso-Saenz, et al., An open IoT platform for the management and analysis of energy data, Future Generation Computer Systems (2017), http://dx.doi.org/10.1016/j.future.2017.08.046.

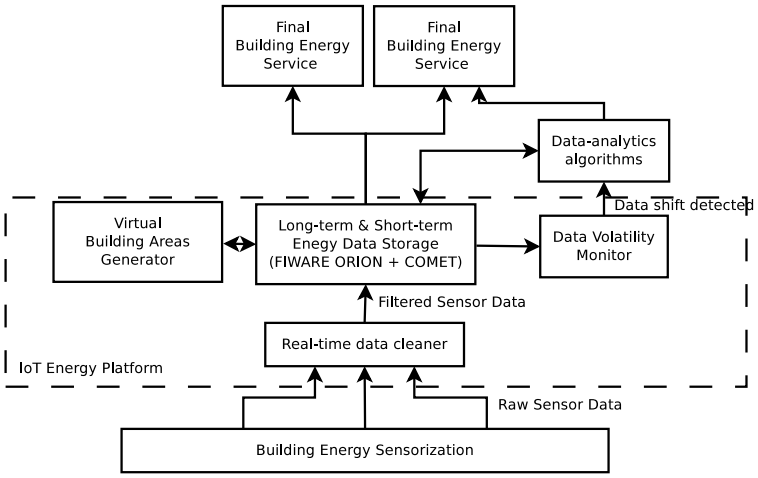


Fig. 1. Conceptual view of the IoT Energy Platform (IoTEP).

consumption using time-driven and event-driven mechanisms for Building Automation and Control Systems. Their Building Energy Management Systems (BEMS) implementation is realized with an IoT platform, introduced by Echelon Corp that includes chips, stacks, communication, application interfaces (API) and management software. Their approaches to calculate the energy demand are based in time (fixed or sliding length of the time windows with the possibility of overlapping) and in events (occupancy).

The MultiAgent System (MAS) named SAVES (Sustainable multiAgent systems for optimizing Variable objectives including Energy and Satisfaction) defined in [21] is used in [22] regarding actual occupant preferences and schedules, actual energy consumption and loss data measured from a real test bed building at the University of Southern California in order to predict energy consumption at different levels (frequency of prediction and device aggregation).

Other works provide energy data management solutions without focusing on analytic aspects. This is the case of the Virtual SCADA architecture for cloud computing (VS-Cloud) that encompasses Cloud Computing for energy data storage [23]. VS-Cloud mainly focuses on the orchestration of components in Smart Grids and the safety storage of sensitive data executed actions, incidents or alarms. Therefore, its domain of application is more related to risk management.

Similarly, the work in [24] proposes an automation platform for energy monitoring. However, such platform does not provide any particular feature to support energy data analytics as it focuses more on the definition of control strategies for energy saving.

Unlike the aforementioned initiatives, our work provides a holistic energy data management and analysis solution. Our platform also follows an open approach by relying on the well-established FIWARE initiative. In that sense, the present work includes explicit features like data volatility monitoring and outliers detection to ease the deployment of data mining algorithms and other services over of the stored data.

FIWARE brings other advantages with respect to previous solutions: firstly, the whole platform orchestration is done by means of lightweight RESTful APIs, that facilitate its further extension; and secondly, the definition of an information model compliant with NGSI standard allows to come up with a homogeneous view of the energy-related data within a building. This feature is key to exploit the potential of gathering energy data. What we propose here is not only an archive of data, but a comprehensive flexible and powerful tool that will serve as the breeding ground for the creation of context-aware tailored energy feedback platforms that could be realized at a scale never considered before, even reaching national levels.

2.2. IoT platforms

The Internet of Things paradigm is the second pillar of this initiative. All the literature indicates that small devices connected to the internet in buildings will be the norm in the near future. With the right algorithms and communication mechanisms, this situation will enable the monitoring and characterization of energy behaviors and energy consumption in buildings.

The need of effective instantiation of IoT under realistic conditions has generated a varied ecosystem of methodologies and tools taking the form of integrated IoT platforms. In that sense, it is possible to find several surveys in the literature that review existing proprietary and open-source platforms in the IoT ecosystem [12,14,25]. Other important aspects like data ownership, security and privacy [26] or data storage [25] have been also deeply studied in the IoT domain. The reader is referred to this sources to expand on the state of the art.

According to such reviews, some relevant IoT platforms follow a similar open-source and centralized approach along with heterogeneous sensor support like IoTEP. This is the case of Nimbits[2] that provides an open source Java library for developing Java, Web and Android solutions to connect to a Nimbits Server. This back-end part enables simple processing of the collected data based on rules. However, it does not comprise any advanced data-analytics support. ThingSpeak[3] features the acquisition, visualization and analysis of data but this is done by means of the proprietary Matlab tool, what may make more difficult the popularization of the platform.

One feature frequently neglected by existing IoT platforms is the support of built-in data mining features able to generate new useful knowledge from the collected and stored data [14]. In real IoT deployments, this processing and analysis task has been frequently done by third-party services. However, integrating certain data mining functionalities as built-in features of platforms would provide a great benefit in a wide range of domains, for example: quick statistics, easy to generate digests or sanity checks. In

2 https://www.nimbits.com/.
3 https://thingspeak.com/.

Please cite this article in press as: F. Terroso-Saenz, et al., An open IoT platform for the management and analysis of energy data, Future Generation Computer Systems (2017), http://dx.doi.org/10.1016/j.future.2017.08.046.

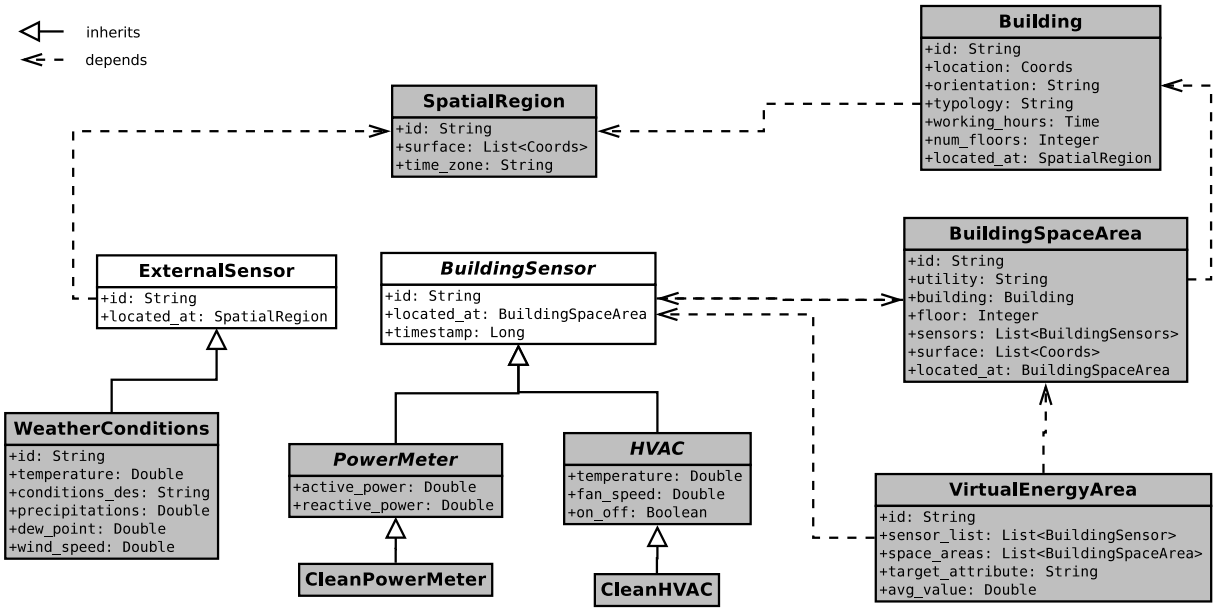


Fig. 2. IoTEP information model.

that sense, only a few IoT platforms actually include native data-analytics features. As a matter of fact, SensorCloud[4] enables a simple interface for common operations like smoothing, filtering and interpolation whereas GroveStreams[5] provides some real-time data analytics mechanisms. However, none of them support sensor heterogeneity nor follow an open source approach like IoTEP does.

As for the energy ecosystem, several research lines have already stated the feasibility and suitability of data analysis in order to increase energy awareness within a building [13]. In that sense, IoTEP provides one of the first steps towards such a data-mining enrichment by providing several features fully focused on easing the analysis of IoT energy data namely, real-time data cleaning, data volatility detection and data reduction procedures.

Finally, our work is enclosed within the FIWARE architecture. The high-level goal of this architecture is to build the Core Platform of the Future Internet, introducing an innovative infrastructure for cost-effective creation and delivery of versatile digital services, providing high QoS and security guarantees. In that sense, FI-LAB [27] conforms live instances of generic enablers, available to developers for free experimentation within this technology.

Some initiatives have started to profit from FIWARE in several domains. One of the most ambitious works is the application on [28] which established a world-wide semantic interoperability solution combining the NGSI, which is part of the core of the FIWARE initiative, and oneM2M context interfaces. Apart from that, [29] demonstrated the suitability of the FIWARE paradigm to compose Future-Internet applications by means of the integration of generic enablers. In a similar manner, [30] put forward a semantic mechanism to integrate data from different types of devices by also using FIWARE components. Finally, in a more functional domain, [31] made use of certain enablers, like ORION context broker, to create a cloud-based gesture recognition application. Also, [32] describes a sensor management for seaports based on the FIWARE platform. It is therefore possible to say that our work seems to be one of the first efforts to make use of FIWARE enablers in the building energy domain, and furthermore in the energy domain in general.

3. IoT Energy Platform (IoTEP)

This section explains in detail the proposed IoTEP solution. Since the management of the energy data is its key feature, we firstly describe the information model used to define all the data within the IoTEP ecosystem; next, we put forward the specific architecture of the platform that deals with the energy data according to the model.

3.1. Information model

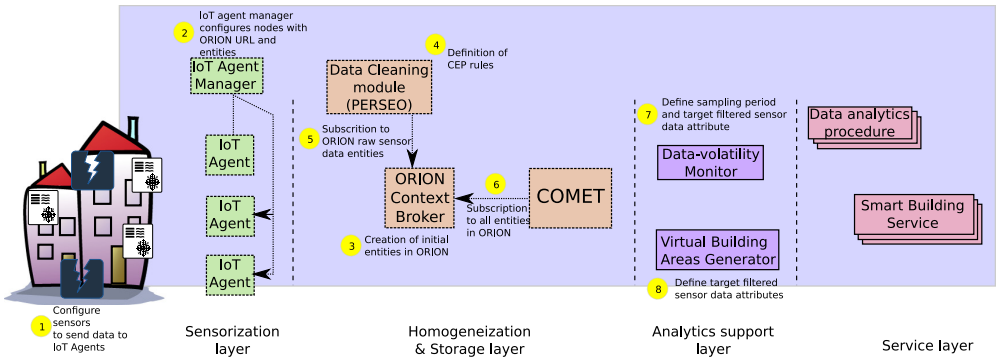
One of the first steps towards the realization of IoTEP was to define a common information model for the whole platform. Such a model must be compliant with the NGSI information model commonly accepted in the FIWARE ecosystem, what facilitates interconnection with other models and other users. This information model follows an entity–attribute approach where entities represent real or virtual elements of interest. Each entity has a type what allows to define type-based hierarchies. In this way, an entity has its own defined attributes and the inherited ones from its ancestors. The IoTEP information model is depicted in Fig. 2. The model design follows the UML class notation with two types of relationships, inheritance and dependence. Each of them is represented by a different arrow in the figure. Whilst inheritance indicates that the child element comprises all the attributes of its parent element, the dependence relationship indicates that an instance of the element at the arrow's origin contains an attribute referencing at one or more instances of the element at the arrow's destination.

Focusing on the content of the model, one can find among its components three key elements related to the energy ecosystem of a building by means of NGSI entities.

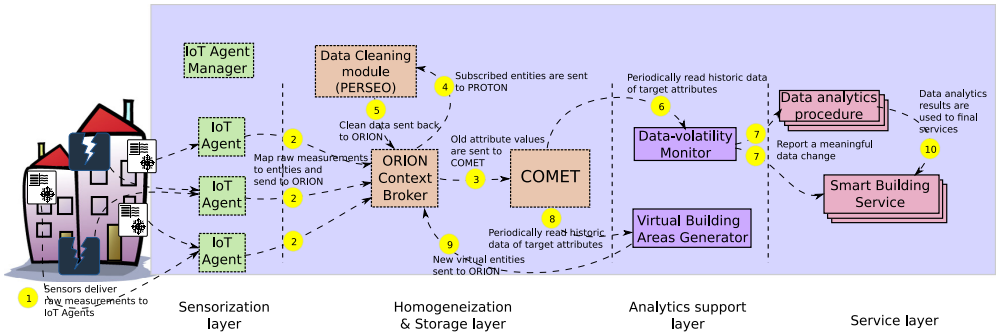
To begin with, the entity *building* models the target building. Several operational and architectonic details of the building are included as attributes on this entity. Examples of information in

[4] http://www.sensorcloud.com/.
[5] https://grovestreams.com/.

Please cite this article in press as: F. Terroso-Saenz, et al., An open IoT platform for the management and analysis of energy data, Future Generation Computer Systems (2017), http://dx.doi.org/10.1016/j.future.2017.08.046.



(a) IoTEP workflow for bootstrap stage.



(b) IoTEP workflow for operational stage.

Fig. 3. Platform general workflow.

this section are: opening hours or building use (e.g., company headquarters, university faculty, etc.) but also physical relevant attributes such as fabrics, windows, orientation, and so forth. Moreover, the *spatial region* entity defines the geographic region containing the building. This entity would help to link together buildings located in similar geographic regions that, as a consequence, might share certain energy-related characteristics. The inner structure of a building is represented with the *building space area* entity. This entity gathers the different spatial areas within a building (e.g., classrooms, corridors, halls, landings, etc.). Furthermore, a recursive structure of these areas can be made with their *located at* attribute to represent, for example, that a classroom is inside a *teaching zone*.

This way of introducing data about the buildings and the spaces will made the communication between a Building Information Modeling (BIM) platforms and the IoTEP platform straight forward, what would facilitate the transfer of information among members of a given team.

The second group of entities refers to the energy sensors deployed in the building and the data they collect. This is modeled by means of the *building sensor*, *power meter* and *hvac* entities. Each entity includes the set of attributes monitored by the corresponding energy sensor along with other metadata (e.g., location of the sensor or timestamp of each observation). The *clean* version of these entities refer to the sensor data generated after the data filtering process as described in Section 3.2.2.

The third group of entities focus on representing sensors that are not necessarily within the infrastructure of the building but that may provide useful when collecting energy data. This is the case, for example, of weather stations reporting conditions of the building site. As Fig. 2 shows, this is defined by means of the *external sensor* and *weather conditions* entities.

Finally, only the entities in gray in Fig. 2 have instances stored in ORION and COMET as we will see later.

3.2. Platform architecture

The proposed IoTEP has been structured in four different layers in an incremental approach (this is shown in Fig. 3). In the upcoming sections, a detailed description of each layer is given.

3.2.1. Sensorization layer

This layer is in charge of connecting physical devices or actuators that are going to provide energy data to the platform. Once this is done, it maps the collected data to the NGSI entities of the information model (described in the previous section) and sends the mapped information to the upper homogenization and storage layer.

For the realization of this layer, we have made use of the FIWARE IoT Agent enabler [33]. In a nutshell, this enabler allows to automatically perform the aforementioned data mapping. Different types of this enabler support transport protocols to connect to the physical devices like MQTT.[6] or Lightweight M2M (LwM2M)[7].

Consequently, during the bootstrapping phase of the platform, a set of IoT Agents are configured with the NGSI entity type associated to each of its associated sensor by means of the IoT Agent Manager (see Fig. 3(a)). In particular, power meters deployed in the target building are mapped to the *power meter* entity type whereas HVAC devices are mapped to the *hvac* one. Furthermore, we developed an ad-hoc agent to parse the weather conditions coming

6 http://mqtt.org.

7 http://openmobilealliance.org/iot/lightweight-m2m-lwm2m/.

Please cite this article in press as: F. Terroso-Saenz, et al., An open IoT platform for the management and analysis of energy data, Future Generation Computer Systems (2017), http://dx.doi.org/10.1016/j.future.2017.08.046.

from an external third-party weather service to the *weather conditions* entity on a regular basis. During the operational phase (see Fig. 3(b)) each time an IoT Agent receives the raw measurements from a physical device, it *inflates* the entity instance associated to the device in upper layer by means of a RESTfull API, in the homogenization and storage layer (will be described in the next section).

3.2.2. Homogenization and storage layer

In this layer, all the collected energy data from the previous layer is conveniently stored in a uniform solution. This way, this layer addresses the heterogeneity of the incoming energy-related data. Moreover, it contains real time data cleaning stage what ensures the quality of the data collected.

Sensor data repository. Regarding the energy-related data storage, this has been achieved by integrating two FIWARE components.

Firstly, ORION context broker [34] implements a publish–subscribe store providing data access by means of the NGSI-10 API [35]. In IoTEP, this enabler stores the entity instances of the information model. By means of the NGSI update operation, IoT Agents in the sensorization layer update the sensor entities' attributes in real time with the new readings from the devices.

Secondly, the COMET enabler [36] is used for supporting access to historic time series data extending the ORION functionality. In that sense, COMET adheres to the same information model, thus, it does not require any further data harmonization process. It incorporates an ad-hoc API to retrieve raw historical sensor data along with several built-in simple aggregation functions over such data (e.g., provide the sum, min or max of the collected observations for a specific time period).

During the bootstrapping phase of the platform, ORION is initiated with the *static* attributes of the entities in the information model (e.g., 'identifier', 'located at' or 'orientation' attributes) and COMET subscribes in ORION to the *dynamic* attributes of the entities to receive each new value (see Fig. 3(a)).

Sensor data cleaning. Concerning the data quality assurance, we developed a data cleaning module to remove the outliers that might be contained in the raw measurements from the sensors. In that sense, outliers have been reported to be the most prominent quality issue of energy data [37,38].

This module had two key requirements. To begin with, the data cleaning process must be done in a timely manner in order to avoid potential bottlenecks. Furthermore, in an IoT ecosystem we should expect a great variety of data formats and structure. Thus, such data cleaning should be done after data homogenization in order to simplify the overall computational cost of the cleaning stage.

In order to cope with the time-processing constraints, we opted for following the Complex Event Processing (CEP) paradigm to develop a real-time data cleaning module. CEP focuses on timely processing streams of information items, so-called events, by filtering, aggregation or pattern discovery using predefined rules following the event–condition–action paradigm [39]. In the present setting, the incoming events are the readings from the energy sensors, the conditions to be detected are whether a reading should be considered or not an outlier and the action of the final insertion of the cleaned data in the storage structure of the platform.

For the outlier definition, we followed a strategy based on quartiles with fences [40]. In brief, such a strategy extracts the median, the lower $\mathcal{Q}_1$ and upper quartiles $\mathcal{Q}_3$ (aka 25th and 75th percentiles) along with the interquartile range $\mathcal{IQ}(= \mathcal{Q}_3 - \mathcal{Q}_1)$ of the data set under study. On the basis of such statistics, two fences are defined,

- Lower outer fence: $\mathcal{Q}_1 - 3 \times \mathcal{IQ}$
- Upper outer fence: $\mathcal{Q}_3 + 3 \times \mathcal{IQ}$

This way, a measurement beyond such fences is considered an *extreme outlier*.

The translation of this strategy to CEP allows to calculate such fences incrementally and update their boundaries each time that a sensor pushes in new data. In particular, two types of CEP rules were defined. The first one comprises the rules in charge of computing for each sensor the aforementioned statistics with respect to each of its parameters. For the sake of clarity, the pseudocode of the CEP rule in charge of calculating the fences for power meter sensors is shown here and it looks as it follows:

```
CONDITION PowerMeter.groupBy(id).within(t_int^clean) as A
ACTION    new PowerMeterStats(A.id,
          calculateLowerOuterFence(A.active_energy))
          calculateLowerOuterFance(A.reactive_energy),
          calculateUpperOuterFence(A.active_energy)
          calculateUpperOuterFence(A.reactive_energy))
```

where `groupBy` and `within` are two sliding windows. While `groupBy` splits the stream of power-meter data with respect each particular device, `within` defines a time window to retain the last power-meter data generated during the last t_{int}^{clean} time units. After this, the action part of the rule, generates a new `power meter stats` event comprising the percentiles for each sensor's attribute considering the data included in the time window. It is important to note that this rule would fire each time that new power meter data is injected into the CEP system.

The second set of rules performs the actual extreme outliers detection. Again, there is one rule per sensor type in charge of this task. The pseudocode of the CEP rule to detect the outliers in the power meter data is shown next,

```
CONDITION PowerMeter as A
          AND PowerMeterStats as B
          AND A.id = B.id
          AND A.active_energy ∈ [B.active_energy_lowerFence,
          B.active_energy_upperFence]
          AND A.reactive_energy ∈ [B.reactive_energy_lowerFence,
          B.reactive_energy_upperFence]
ACTION    new CleanPowerMeter(A.id, A.timestamp, A.located_at,
          A.active_energy, A.reactive_energy)
```

Describing it briefly, this rule fires each time that a new power-meter reading is received. The condition part of the rule matches such reading with its associated statistics and checks whether each parameter is contained in its own fences. If that is the case, the reading is considered that has been cleaned. As a result, the action part creates a new `clean power meter` event with the pre-processed data.

A very similar approach is followed for the HVAC data but, this time, using the thermostat temperature attribute of this type of sensor in order to give rise to `clean hvac` events.

The implementation of this CEP mechanism has been made with the Perseo FIWARE enabler [41]. This component incorporates a CEP engine and an SQL-based event processing language to define and execute the CEP rules. Furthermore, it leverages the publish-subscription capabilities of ORION. This way, the engine receives each entity instance, which data has been just updated in ORION, as incoming events; and the cleaned events generated by the rules, automatically update their associated entities in ORION (Fig. 3(b)). Hence, during the bootstrapping phase (see Fig. 3(a)) this component is configured with the rules to be executed and the list of entities in ORION to subscribe (in this case, *power meter* and *hvac* entities).

Please cite this article in press as: F. Terroso-Saenz, et al., An open IoT platform for the management and analysis of energy data, Future Generation Computer Systems (2017), http://dx.doi.org/10.1016/j.future.2017.08.046.

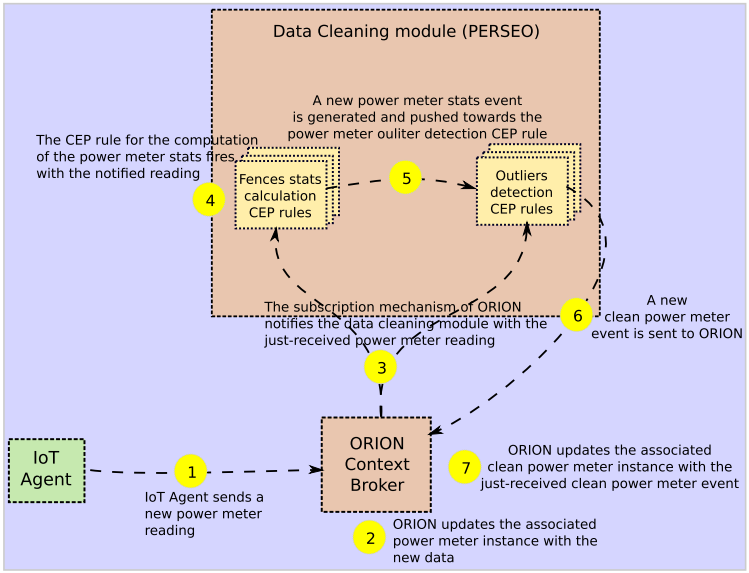


Fig. 4. Workflow of the cleaning of power meter readings.

Finally, Fig. 4 shows an illustrative example of the workflow of the CEP cleaning mechanism and its connection with the sensor data repository. As this figure depicts, each raw sensor reading coming from the IoT Agents is initially stored in ORION by updating its associated *building sensor* instance. In the figure's scenario, a new power-meter reading will update the *power meter* instance representing the sender's sensor (steps 1 and 2 in the figure).

Next, ORION automatically notifies to the data cleaning module the new reading (step 3). This notification fires the two types of CEP rules described before (steps 4 and 5). At the end, the module outcome takes the form of a `clean power meter` event that updates the associated *clean power meter* instance in ORION. This *clean power meter* instance represents the cleaned version of the power meter sensor updated in step 2. Moreover, we should note that all the aforementioned interactions occur following a push-style communication enabling the real-time processing.

3.2.3. Analytics support layer

The third layer of the platform embraces all the functionalities of the platform to provide support for data mining services that can run on top of the platform. In particular, two features have been included in this layer, an energy data volatility detector and a virtual entities generator.

Virtual energy building areas generator (VEBAG). The amount of data that we are able to collect in smart buildings by means of large sensor networks sometimes does not increase the *information volume* because of redundancy. Depending on its nature, this redundancy is treated using different approaches: redundancy detection, data compression, feature extraction, and some others [42].

IoTEP works under the hypothesis that a clever way to reduce the number of variables taking part in the models can not only decrease the computation costs but also increase the accuracy on predictions and classification. In this way, the creation of abstract entities will be justified from the data analytics side, based on the assumption of the existence of this redundancy.

Therefore, the goal of the VEBAG module is the creation of high level entities that preserve as much information as possible in the data set but yet, reducing the volume of it. In this case, we want to create virtual areas comprising several *building space areas*, finding patterns in the energy-related use and defining these virtual areas according to such information to optimize the content of information.

To do so, we aggregate each attribute per energy device daily. This aggregation can be easily done with the built-in RESTful aggregation functions provided by COMET within the homogenization and storage layer. That way, we can represent each device as a time series having one attribute measurement per day and with this, it is possible to find a clustering algorithm that groups every attribute of the time series finding some distinctions between them, like DBSCAN or longitudinal k-means.

Once every device is assigned to a cluster or virtual area, the generator computes the mean of the elements of each cluster to get an average measurement. Finally, each generated cluster is stored in the storage layer as an instance of the *virtual energy area* entity (see Fig. 3(b)). In that sense, this generator is launched on a regular basis or when certain data shifts are detected in the data by the data volatility monitor (described in the next section). Fig. 5 depicts an illustrative example of this process given the building's floor.

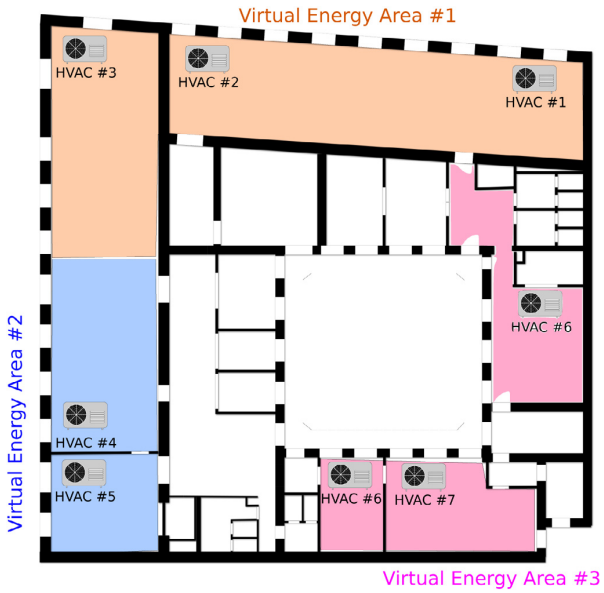
Firstly, Fig. 5(a) shows the distribution of room-based building space areas along with their HVACs. It should be recalled that each of these areas and sensors will be stored as different instances in IoTEP. Furthermore, the figure also shows an example of a possible time-series plot of the regulated temperature for each HVAC for illustration purposes.

Next, Fig. 5(b) shows the *virtual energy areas* generated on the basis of the aforementioned temperature time series. As we can see, the six initial room-based building space areas have been merged into three instances of *virtual energy areas* by grouping together the HVACs with similar time series. This way, rooms 4, 5 and 6 and their associated HVACs have been merged into a single area (*virtual energy area 3* in the figure).

All in all, the generation of these virtual energy areas enables the platform to provide multiple views of the energy status of a building. In a low-level setting, we can monitor energy parameters from a single-sensor point of view. Over such simple view, we can also extract energy parameters related to a particular building spatial area (e.g., room, corridor and the like) by simple aggregation

Please cite this article in press as: F. Terroso-Saenz, et al., An open IoT platform for the management and analysis of energy data, Future Generation Computer Systems (2017), http://dx.doi.org/10.1016/j.future.2017.08.046.

(a) HVACs and room-based building space areas.



(b) Virtual Energy Areas generated based on the HVACs' temperature time-series.

Fig. 5. Example of generation of *virtual energy areas* considering the HVAC temperature in a building floor.

using the *building spatial area* instances. Finally, *virtual energy area* instances enrich the energy awareness by providing an extra layer of perception that is not constrained by the building architectural structure. This way, it is possible to monitor building areas with similar energy behaviors simultaneously.

Data volatility monitor. In order to come up with real energy-aware services, the monitoring of certain energy parameters of a building becomes paramount. This includes detecting either abnormal energy consumption related to building spaces or an abnormal temperature setting related to HVACs.

Please cite this article in press as: F. Terroso-Saenz, et al., An open IoT platform for the management and analysis of energy data, Future Generation Computer Systems (2017), http://dx.doi.org/10.1016/j.future.2017.08.046.

For that goal, the data volatility monitor focuses on computing the current rate of change of each energy sensor parameter included in the storage layer. This is done in three steps.

Firstly, we extract the historic data set of the target energy parameter for a particular sensor with respect to a pre-defined time period t_{int}^{vol} from COMET. Then, the average rate of change among pairs of consecutive observations of the attribute is computed. Finally, if such averaged value is substantially different that the historic rate of change of that attribute then an alarm is triggered. For the sake of clarity, the pseudo-code of this process is shown in Algorithm 1.

Algorithm 1: Data volatility calculation.

Input: Type, identifier and energy parameter of the monitored sensor ($sensor_{type}$, $sensor_{id}$, $sensor_{attr}$), time interval under study (t_{int}^{vol}) and historic rate of change of the considered parameter for the target sensor (rh_{attr}^{sensor}).
Output: Data volatility alarm, if any.

```
/* Historic data extraction                                   */
D ← get_COMET_raw_historic_data(sensor_type, sensor_id, sensor_attr, t_int^vol)
/* Average data-rate change calculation                       */
d_prev ← 0 r_avg ← 0 n ← 0
for each d ∈ D do
    r ← |d − d_prev|
    r_avg ← r_avg + (d − r_avg)/n
    d_prev ← d n ← n + 1
/* Meaningful data-rate change detection                      */
if r_avg >> rh_attr^sensor then
    return data_volatility_alarm(sensor_type, sensor_id, sensor_attr, r_avg)
```

This alarm is received by the final energy services on top of the platform and the VEBAG module. If this module receives a set of consecutive alarms related to the same energy parameter in a short period of time then it might indicate that the energy similarities in between building areas have changed. In order to capture such shift, VEBAG re-launches the clustering process to reconfigure the virtual areas related to such energy factor. In that sense, this monitor is endlessly executed every t_{int}^{vol} time units in order to keep a continuous control over the sensor data streams.

Finally, we would like to notice that this last mechanism along with the CEP data cleaning described in Section 3.2.2 might provide some clues to building operators about data inconsistencies due to sensor interferences. In particular, the data cleaning module can remove readings that are not consistent with the normal operation of a sensor whereas the data volatility mechanism can also detect abnormal disturbances in the data rate change of a sensor reporting that something unusual is happening.

3.2.4. Service layer

Although not that central when considering the architecture of the platform here developed, the Service Layer is the last level of the IoTEP. This layer serves as interface between the IoTEP and the user, that could be anything from a building services manager to the back end of a smartphone application.

At this level, the data analytics procedures can be invoked and their results visualized. Also, smart-building services that may be the norm when the smart-building paradigm is fully established will be nested at this level of the IoTEP platform, and will allow features such as advanced HVAC predictive control, home automation, fuel poverty evaluation, sick building syndrome diagnostics, risk situations for vulnerable people (as in heat waves), smart tariff strategies, and many others.

4. Validation of the platform

In order to test the feasibility of the proposed platform, IoTEP has been instantiated in a real pilot that allowed us to evaluate functionalities of the new platform. Here we provide some details of the evaluation scenario.

4.1. Pilot description

IoTEP was instantiated at the University of Murcia, Spain. During the last three years, this university has carried out an ambitious plan to monitor and control its buildings' infrastructures distributed across the university premisses. The number of buildings monitored and the automated services have increased quickly in the last years, what serves well the purpose of testing the plasticity of the platform presented in this paper. It should be noted that the sensorization of the buildings at the University of Murcia was done independently of this project, so the fact that the platform was able to allocate the data coming from all the sensors was already a proof of its validity.

In this context, IoTEP was used as the main enabler of an energy efficiency campaign at three cases, namely the Faculty of Chemistry and two multi-disciplinary research and technological transfer centers within the university. Details of the three buildings are provided in Table 1.

Lastly, the evaluation of IoTEP covered a three-month winter campaign from 01/10/2016 to 28/02/2017.

Platform configuration. IoTEP was installed in a centralized server with CentOS 6.7 as operating system, 8 GB RAM and 250 GB hard disk. Besides, Table 2 sums up the configuration of the inner parameters of the platform. It should be reminded that t_{int}^{clean} defines the time interval used by the CEP cleaning mechanism to compose the quartile fences (Section 3.2.2) whereas t_{int}^{vol} indicates the length of the time series considered by the data volatility mechanism to infer meaningful data shifts (see Algorithm 1).

Before the deployment of IoTEP in the pilot, a full covering of energy related variables was done in the buildings under study. After preliminary evaluations, it was discovered that there are three families of data that are fundamental to understand the energy behavior of the building users and heat losses of the envelopes. The three families are: building characteristics, energy streams and building state.

The building characteristics are the physical description of the building. Detailed blueprints of the building were obtained from the department of estates of the university together with detailed plans of constructions. This information together with visual inspections carried out by the members of our team have allowed as to have a rather full description of the condition of the building thermal envelope. With this, it was possible to use building physical models to analyze and predict the heat flows of the building and therefore the energy performance of the fabrics.

About the second family, we were able to monitor in real time with a sampling period of 10 min the operation of more than 200 conditioning units in real time. This included the status of the machines (on/off) and the set point temperatures. It was also possible to obtain the technical characteristics of the machines, what together with the rest of the data allowed us to have a rather accurate proxy of specific power consumption in real time. To contextualize this individual power consumption, the total power consumption of the building was also measured.

Finally, it was needed to know what the conditions on the interior of the spaces of the building were. For this, we monitored in real time the temperature of more than 200 spaces. This temperatures are in accordance with the data taken from the conditioning systems what allowed us to create virtual control volumes/zones in which to evaluate energy flows.

Please cite this article in press as: F. Terroso-Saenz, et al., An open IoT platform for the management and analysis of energy data, Future Generation Computer Systems (2017), http://dx.doi.org/10.1016/j.future.2017.08.046.

Table 1
Use case building characterization.

	Faculty of chemistry (FC)	Technological transfer center (TTC)	Research center (RC)
Location (coords)	38.02, −1.16	37.72, −1.09	38.02, −1.17
Orientation	south-west	south-west	south-west
Surface area	1500 m^2	3323 m^2	1000 m^2
Floors	6	4	2

Table 2
IoTEP parameters setting.

Parameter	Description	Value
t_{int}^{clean}	Time window length for sensor stream fence calculation	30 days
t_{int}^{vol}	Time period for data volatility calculation	2 hours

Table 3
Information model entities distribution per building.

Entity	Number of instances		
	FC	TTC	RC
Spatial region	1	1	1
Building	1	1	1
Building space area	344	16	10
HVAC	239	0	4
Clean HVAC	239	0	4
Power meter	1	13	4
Clean power meter	1	13	4
Weather conditions	1	1	1

The IoTEP was created in such manner that it allows to allocate all this information in two ways: in the form of data stream, and in the form of "static" information. In this way, the description of the building is allocated on the *building* entity previously described. The characteristics of the conditioning system and the data stream can be placed on the *HVAC* and *power meter* entities created for this purpose.

This comprehensive set-up fully monitors the most important energy related aspects of the building, what could be a two-bladed sword. In principle, this allows to do high level reasoning on the data with the high added value that this represents; however, such a large flow of data may render the infrastructure slow and inefficient with such a heterogeneous data. With the solution proposed in this paper we overcome the problems, leading to a platform that, because of the efficient handling of data inherited from FIWARE, allows for the true real time comprehensive data analysis of buildings. With the advantages that this represents.

As a result of this study, Table 3 shows the distribution of instances of the entities of the IoTEP information model stored in ORION per building.

4.2. Pilot objectives

The goal for this testing campaign was to develop a new service able to predict the next-day energy consumption of each of the three buildings, and with this to evaluate the framework we present at all the different levels. However, it should be reminded that this is only an example of the variety of features that could be implemented on IoTEP. The service tested would be instrumental for the department of estates of the university in order to plan energy-saving actions and advanced versions of model predictive control.

As Fig. 6 shows, this service was developed on top of IoTEP i.e. on the service layer shown in Section 3.2.4, by using its functionalities. It was implemented as a web application allowing the control of some of the IoTEP features by the buildings manager to carry on decisions according to data analysis results. Consequently, this application acts as a dashboard that allows users to control

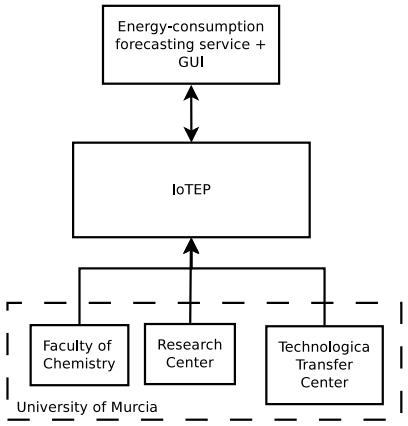


Fig. 6. Representation of the IoTEP pilot evaluation.

the platform and access the aforementioned energy consumption service (see Fig. 7).

In terms of access of the inner features of IoTEP the application includes the following actions,

- Firstly, it is possible to visualize the most recent readings of the HVAC devices per each room of the building. For this feature, the application makes use of the ORION component of the platform.
- Secondly, it is also possible to visualize the HVAC data given a time range defined by the user. For this purpose, the application leverages the raw historic data extraction method of COMET.
- Moreover, this dashboard also allows to control and visualize the results of the *virtual energy areas* generation of the platform (VEBAG module). In that sense, the user can also select the clustering method, and the number of clusters will be selected automatically by the Calinski–Harabasz index.

Finally, the energy consumption prediction service was also integrated in this application. On this way, building managers have full control over all the data analytic process starting from data visualization, aggregation and clustering to the final energy prediction procedure. This integration allows to perform such prediction for several granularity levels targeting from single devices, space areas or *virtual energy areas*. This multi-faced prediction is a key innovation aspect of the application.

For the evaluation of the platform, we studied the suitability and feasibility of the multi-layered view of the energy-related information proposed by IoTEP by means of the *virtual energy areas* generation. Additionally, we also studied the accuracy of the

Please cite this article in press as: F. Terroso-Saenz, et al., An open IoT platform for the management and analysis of energy data, Future Generation Computer Systems (2017), http://dx.doi.org/10.1016/j.future.2017.08.046.

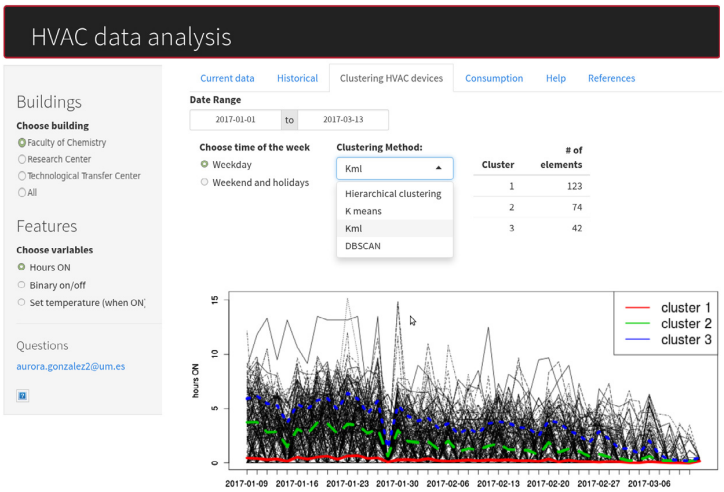


Fig. 7. IoTEP dashboard and energy consumption prediction service.

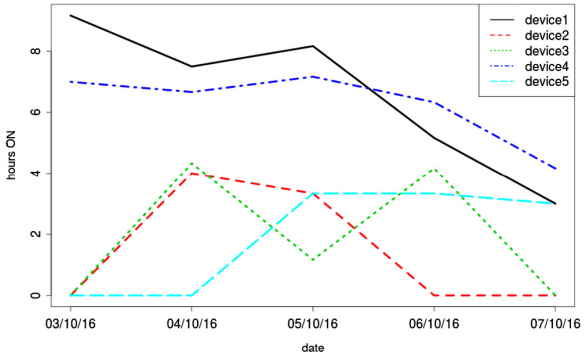


Fig. 8. Time series of 5 HVAC devices.

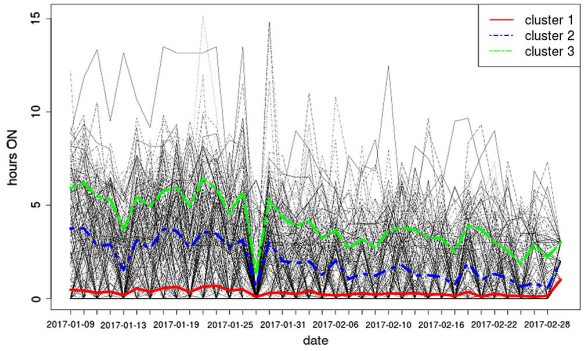


Fig. 9. Cluster evolutions.

energy prediction service when such areas are included as the target entities.

For the generation of these areas, the daily aggregation of data made by the VABAG module was based on counting the hours that each device is tuned on during the day (24 h). As an example, the number of hours that five devices were on during five days is shown in Fig. 8

For the clustering of such aggregated data, we relied on the k-means algorithm [43], but as mentioned before, more algorithms can be used for this purpose. We arbitrarily selected 3 clusters, but a different number can be selected if needed. In Fig. 9 we show the three evolutions of the groups of HVAC within FC that this algorithm identified for working days during the period of study. That way, we found rooms in this building with high use pattern (cluster 3, comprising 47 devices), rooms with little use (cluster 1 with 118 HVACs) and rooms presenting an intermediate frequency of use of the HVAC system (cluster 2 with 74 HVACs). The separation of these clusters could be the first step to an intervention strategy to modify the behavior of big consumers.

In the same way, and looking at the infrastructure level, we represent 239 values taken from the HVAC devices into 3 variables providing a 98.7% reduction of data.

Regarding the energy-prediction service, it makes its prediction according to the previous HVAC grouping within FC. Hence, we compare its performance with the use of the raw data set and in combination with environmental variables. Being the inputs and outputs of the model identified, we followed the next steps [44]: Being the inputs and outputs of the model identified, we followed the next steps [44]:

1. Standardization of inputs
2. Splitting the data into training (75%) and test set (25%)
3. Validation: 10-fold cross validation and 5 repetitions over the training data set using several models: random forest, artificial neural networks and support vector regression.
4. Evaluation: Using the RMSE metric to evaluate the models and its coefficient of variation for comparison.

The scenarios to compare are based on the different inputs to consider:

- "Hours on" average per cluster of the previous day
- Weather predictions from Weather Underground API.[8]
- Raw HVAC data (every HVAC device daily usage)
- Both average per cluster and weather predictions

As we can see in Table 4, with a really reduced number of inputs (only 3 variables), for every model we obtain very good results compared to the others. That way, the use of clusters for creating

[8] https://www.wunderground.com/.

Please cite this article in press as: F. Terroso-Saenz, et al., An open IoT platform for the management and analysis of energy data, Future Generation Computer Systems (2017), http://dx.doi.org/10.1016/j.future.2017.08.046.

Table 4
RMSE (and CV-RMSE) of the different models and inputs.

Model	HVAC clusters	Weather	Raw HVAC	Clust + Weath
RF	0.32 (10.53)	0.513 (17.74)	0.358 (11.83)	0.356 (11.76)
SVM	0.316 (11.03)	0.635 (22)	0.446 (14.76)	0.461 (15.23)
BRNN	0.281 (9.48)	0.423 (14.63)	0.347 (11.47)	0.398 (13.15)
# Inputs	**3**	23	239	26

higher level entities is proved to be useful. Although this is a rather arbitrary method, we prove with this that the platform serves to host algorithms for data analysis and prediction on a very versatile way

Comparative results. In the work [22], CV-RMSE is used in order to validate their results. They are evaluating both aggregated (total) and disaggregated (cooling and ventilating) energy consumption in a daily, weekly and monthly basis. When we compare our results with theirs, we are obtaining 6% less of variance for the RMSE, which is very satisfactory.

In addition, the Recommended Values for Baseline Model from ASHRAE Guideline 14 [45] account for the CV-RMSE smaller than 30% for daily predictions which we reach with ease (our best performance returns a 9.48 %, see Table 4).

To sum up, with this small example we show what can be implemented on the service layer of the IoTEP. With this, we intend to prove how rather complex methods can be implemented on a simple way in our platform. Also, we have shown an example of reducing data volume taking advantage of data redundancy reduction doing clustering. For this specific example we have taken three clusters as an arbitrary number and we have shown that total energy can be predicted with them. This was done as it evaluates all the features of the platform that we show in this paper, but many other applications and examples can be developed following the principles shown in Section 2.

4.3. Lessons learnt

From this first deployment of IoTEP, we can draw up some remarks.

Firstly, the results of the preliminary sensorization study of pilot were easily integrated in the IoTEP information model. This allowed to homogenized all such results in a common format and showed the versatility of the model.

Secondly, the integration of data mining support procedures as part of the platform made possible the easy development of a final service for energy data mining. In that sense, developers only needed to focus on the actual functionality of the service related to the prediction algorithms since other important tasks of the data analysis like data pre-processing or clustering were already provided by the platform.

Finally, the idea of providing a multi-layered view of the energy status of a building by means of clustering techniques has proved its suitability in the energy prediction service in two aspects. From a data-mining point of view, it reduces the redundancy of data and, thus, making up lightweight models. From a more functional point of view, the level of abstraction that the virtual energy areas provide might help building managers to better understand certain energy behaviors within the building.

All in all, this pilot has helped us to confirm that the integration of data analytics support features as part of the IoT platform is currently a key requirement in the energy domain. This enables the development of more sophisticated energy-aware services in a fast-pace process what seem to be the next natural step towards a more efficient energy-literate society.

5. Conclusions

Due to the importance of the building sector in the end-use energy consumption, it becomes a foremost task to achieve meaningful energy savings that will reduce this energy use in reality.

Despite the fact that IoT technologies have been widely used for the realization of the smart building concept, the simple sensorization of buildings is not enough to make a housing stock that consumes fewer energy resources a reality. IoT is also required to properly process, manage and, above all, analyze the energy-related data that would help to develop final energy-aware services targeting the energy efficiency goal.

In this context, several multi-purpose IoT platforms already provide generic solutions to manage IoT data. However, there is a lack of platforms in this field focusing on (1) the household energy domain and (2) providing support for data analytics. As a result, the present work shows an IoT Energy Platform (IoTEP) that covers the two aforementioned needs by following an open approach based on FIWARE enablers. IoTEP provides several functionalities oriented to the data analytics domain like the CEP data cleaning module or the times series storage along with functionalities for the correct energy management like the data volatility monitoring or the virtual energy areas detector that will allow with personalized energy feedback for the improvement of energy behavior.

Lastly, the platform has been instantiated in a real use case having a large energy sensor network. In that sense, one of the key novelties of IoTEP is that the virtual areas detection has proved to be of great help when it comes to develop an end-use energy prediction service over the platform, but many other services could be implemented with trivial computational effort under this paradigm.

Regarding further work, IoTEP has been developed re-using several open source components that are orchestrated following lightweight RESTfull calls what allows other scientists and engineers to contribute to this platform, opening the door to crowd sourced development. Consequently, new modules and enablers can be smoothly integrated in the existing architecture. In that sense, the integration of other types of sensing approaches beyond mote-class sensors, like crowdsensing, it foreseen as future actions in the platform. This would allow to capture and analyze other forms of human behavior also relevant for the building energy domain.

Acknowledgments

This paper has been made possible thanks to the support of the European Commission through the H2020-ENTROPY-649849, the Spanish National Project CI-CYT EDISON (TIN2014-52099-R) and MINECO TIN2014-52099-R project (grant BES-2015-071956) granted by the Ministry of Economy and Competitiveness of Spain (including ERDF support). Ramallo-González would like to thank the program Saavedra Fajardo (grantnumber220035/SF/16) funded by Consejería de Educación y Universidades of CARM, via Fundación Séneca-Agencia de Ciencia y Tecnología de la Región de Murcia.

References

[1] Odyssee Mure project, Energy efficiency trends in buildings in the EU, lessons from the ODYSSEE MURE project, 2012. http://www.odyssee-mure.eu/publications/br/energy-efficiency-in-buildings.html.

[2] Buildings Energy Data Book, Buildings share of U.S. primary energy consumption stats. http://buildingsdatabook.eren.doe.gov/TableView.aspx?table=1.1.3.

[3] GeSI, GeSI smarter2020 report. http://gesi.org/SMARTer2020.

[4] D. Coley, T. Kershaw, M. Eames, A comparison of structural and behavioural adaptations to future proofing buildings against higher temperatures, Build. Environ. 55 (2012) 159–166. http://dx.doi.org/10.1016/j.buildenv.2011.12.011. http://www.sciencedirect.com/science/article/pii/S0360132311004276.

[5] Project iNSPiRe - Development of Systemic Packages for Deep Energy Renovation of Residential and Tertiary Buildings including Envelope and Systems, 2014. http://inspirefp7.eu.

[6] M. Vellei, S. Natarajan, B. Biri, J. Padget, I. Walker, The effect of real-time context-aware feedback on occupants heating behaviour and thermal adaptation, Energy Build. 123 (2016) 179–191. http://dx.doi.org/10.1016/j.enbuild.2016.03.045. http://www.sciencedirect.com/science/article/pii/S0378778816301992.

[7] J.A. Stankovic, Research directions for the internet of things, IEEE Internet of Things J. 1 (1) (2014) 3–9. http://dx.doi.org/10.1109/JIOT.2014.2312291.

[8] S. Darby, Smart metering: What potential for householder engagement? Build. Res. Inf. 38 (5) (2010) 442–457. http://dx.doi.org/10.1080/09613218.2010.492660.

[9] V. Desley, B. Laurie, M. Peter, The effectiveness of energy feedback for conservation and peak demand: A literature review, Open J. Energy Effic. 2013 (2013).

[10] M.V. Moreno, A.F. Skarmeta, L. Dufour, D. Genoud, A.J. Jara, Exploiting IoT-based sensed data in smart buildings to model its energy consumption, 2015 IEEE International Conference on Communications, ICC, 2015, pp. 698–703. http://dx.doi.org/10.1109/ICC.2015.7248403.

[11] N. Simcock, S. MacGregor, P. Catney, A. Dobson, M. Ormerod, Z. Robinson, S. Ross, S. Royston, S.M. Hall, Factors influencing perceptions of domestic energy information: Content, source and process, Energy Policy 65 (2014) 455–464. http://dx.doi.org/10.1016/j.enpol.2013.10.038. http://www.sciencedirect.com/science/article/pii/S0301421513010604.

[12] M. Zdravković, M. Trajanović, J. Sarraipa, R. Jardim-Gonçalves, M. Lezoche, A. Aubry, H. Panetto, Survey of Internet-of-Things platforms, in: 6th International Conference on Information Society and Techology, ICIST 2016, Vol 1, ISBN: 978-86-85525-18-6, 2016, pp. 216–220, Kopaonik, Serbia. https://hal.archives-ouvertes.fr/hal-01298141.

[13] M. Molina-Solana, M. Ros, M.D. Ruiz, J. Gmez-Romero, M. Martin-Bautista, Data science for building energy management: A review, Renew. Sustain. Energy Rev. 70 (2017) 598–609. http://dx.doi.org/10.1016/j.rser.2016.11.132. http://www.sciencedirect.com/science/article/pii/S1364032116308814.

[14] J. Mineraud, O. Mazhelis, X. Su, S. Tarkoma, A gap analysis of internet-of-things platforms, Comput. Commun. 89 (2016) 5–16.

[15] K. Zhou, C. Fu, S. Yang, Big data driven smart energy management: From big data to big insights, Renew. Sustain. Energy Rev. 56 (2016) 215–225. http://dx.doi.org/10.1016/j.rser.2015.11.050. http://www.sciencedirect.com/science/article/pii/S1364032115013179.

[16] Y. Simmhan, S. Aman, A. Kumbhare, R. Liu, S. Stevens, Q. Zhou, V. Prasanna, Cloud-based software platform for big data analytics in smart grids, Comput. Sci. Eng. 15 (4) (2013) 38–47. http://dx.doi.org/10.1109/MCSE.2013.39.

[17] A. Kumbhare, Y. Simmhan, V. Prasanna, Cryptonite: A secure and performant data repository on public clouds, 2012 IEEE Fifth International Conference on Cloud Computing, 2012, pp. 510–517. http://dx.doi.org/10.1109/CLOUD.2012.109.

[18] A. Ishii, T. Suzumura, Elastic stream computing with clouds, 2011 IEEE 4th International Conference on Cloud Computing, 2011, pp. 195–202. http://dx.doi.org/10.1109/CLOUD.2011.11.

[19] N. Vastardis, M. Kampouridis, K. Yang, A user behaviour-driven smart-home gateway for energy management, J. Ambient Intell. Smart Environ. 8 (6) (2016) 583–602.

[20] A. Ożadowicz, J. Grela, An event-driven building energy management system enabling active demand side management, in: Event-Based Control, Communication, and Signal Processing, EBCCSP, 2016 Second International Conference on, IEEE, 2016, pp. 1–8.

[21] L. Klein, J.-y. Kwak, G. Kavulya, F. Jazizadeh, B. Becerik-Gerber, P. Varakantham, M. Tambe, Coordinating occupant behavior for building energy and comfort management using multi-agent systems, Autom. Constr. 22 (2012) 525–536.

[22] N. Li, J.-y. Kwak, B. Becerik-Gerber, M. Tambe, Predicting HVAC energy consumption in commercial buildings using multiagent systems, Proceedings of the 30th International Symposium on Automation and Robotics in Construction and Mining, ISARC, 2013.

[23] C. Alcaraz, I. Agudo, D. Nunez, J. Lopez, Managing incidents in smart grids a la cloud, 2011 IEEE Third International Conference on Cloud Computing Technology and Science, 2011, pp. 527–531. http://dx.doi.org/10.1109/CloudCom.2011.79.

[24] M.V. Moreno, B. beda, A.F. Skarmeta, M.A. Zamora, How can we tackle energy efficiency in iot basedsmart buildings? Sensors 14 (6) (2014) 9582–9614. http://dx.doi.org/10.3390/s140609582. http://www.mdpi.com/1424-8220/14/6/9582.

[25] A. Botta, W. de Donato, V. Persico, A. Pescap, Integration of cloud computing and Internet of Things: A survey, Future Gener. Comput. Syst. 56 (2016) 684–700. http://dx.doi.org/10.1016/j.future.2015.09.021. http://www.sciencedirect.com/science/article/pii/S0167739X15003015.

[26] J. Zhou, Z. Cao, X. Dong, A.V. Vasilakos, Security and privacy for cloud-based IoT: Challenges, IEEE Commun. Mag. 55 (1) (2017) 26–33. http://dx.doi.org/10.1109/MCOM.2017.1600363CM.

[27] T. Zahariadis, A. Papadakis, F. Alvarez, J. Gonzalez, F. Lopez, F. Facca, Y. Al-Hazmi, FIWARE Lab: Managing resources and services in a cloud federation supporting future internet applications, IEEE/ACM 7th International Conference on Utility and Cloud Computing, 2014, pp. 792–799. http://dx.doi.org/10.1109/UCC.2014.129.

[28] E. Kovacs, M. Bauer, J. Kim, J. Yun, F. Le Gall, M. Zhao, Standards-based worldwide semantic interoperability for IoT, IEEE Commun. Mag. 41 (2016).

[29] S. Sotiriadis, E.G.M. Petrakis, S. Covaci, P. Zampognaro, E. Georga, C. Thuemmler,An architecture for designing Future Internet (FI) applications in sensitive domains: Expressing the software to data paradigm by utilizing hybrid cloud technology, 13th IEEE International Conference on BioInformatics and BioEngineering 2013, pp. 1–6. http://dx.doi.org/10.1109/BIBE.2013.6701578.

[30] F. Ramparany, F.G. Marquez, J. Soriano, T. Elsaleh, Handling smart environment devices, data and services at the semantic level with the FI-WARE core platform, 2014 IEEE International Conference on Big Data, Big Data, 2014, pp. 14–20. http://dx.doi.org/10.1109/BigData.2014.7004417.

[31] A. Preventis, K. Stravoskoufos, S. Sotiriadis, E.G.M. Petrakis, Personalized motion sensor driven gesture recognition in the FIWARE cloud platform, 14th International Symposium on Parallel and Distributed Computing, 2015, pp. 19–26. http://dx.doi.org/10.1109/ISPDC.2015.10.

[32] P. Fernndez, J.M. Santana, S. Ortega, A. Trujillo, J.P. Surez, C. Domnguez, J. Santana, A. Snchez, Smartport: A platform for sensor data monitoring in a seaport based on fiware, Sensors 16 (3) (2016). http://dx.doi.org/10.3390/s16030417. http://www.mdpi.com/1424-8220/16/3/417.

[33] Telefonica I+D, IoT Agent documentation, 2017. http://fiware-iot-stack.readthedocs.io..

[34] Telefonica I+D, ORION context broker documentation, 2017. http://fiware-orion.readthedocs.io.

[35] Open Mobile Alliance (OMA) Specification, NGSI Context Management, 2010. http://www.openmobilealliance.org/release/NGSI/V1_0-20100803-C/OMA-TS-NGSI_Context_Management-V1_0-20100803-C.pdf.

[36] Telefonica I+D, COMET documentation, 2017. http://fiware-sth-comet.readthedocs.io..

[37] W. Chen, K. Zhou, S. Yang, C. Wu, Data quality of electricity consumption data in a smart grid environment, Renew. Sustain. Energy Rev. 75 (2017) 98–105. http://dx.doi.org/10.1016/j.rser.2016.10.054. http://www.sciencedirect.com/science/article/pii/S1364032116307109.

[38] A. Ramallo-González, New method to reconstruct building environmental data, in: Buildign Simulation International Conference BS2015, University of Bath, 2015.

[39] O. Etzion, P. Niblett, Event processing in action, first ed., Manning Publications Co., Greenwich, CT, USA, 2010.

[40] NIST/SEMATECH, e-Handbook of Statistical Methods, 2012. http://www.itl.nist.gov/div898/handbook/.

[41] Telefonica I+D, PERSEO official repository, 2017. https://github.com/telefonicaid/perseo-fe..

[42] H. Hu, Y. Wen, T.S. Chua, X. Li, Toward scalable systems for big data analytics: A technology tutorial, IEEE Access 2 (2014) 652–687. http://dx.doi.org/10.1109/ACCESS.2014.2332453.

[43] C. Genolini, B. Falissard, KmL: k-means for longitudinal data, Comput. Stat. 25 (2) (2010) 317–328. http://dx.doi.org/10.1007/s00180-009-0178-4.

[44] A. Gonzlez-Vidal, V. Moreno-Cano, F. Terroso-Senz, A.F. Skarmeta, Towards energy efficiency smart buildings models based on intelligent data analytics, Procedia Computer Science 83 (2016) 994–999. http://dx.doi.org/10.1016/j.procs.2016.04.213. The 7th International Conference on Ambient Systems, Networks and Technologies, ANT 2016 / The 6th International Conference on Sustainable Energy Information Technology, SEIT-2016 / Affiliated Workshops. http://www.sciencedirect.com/science/article/pii/S1877050916302460.

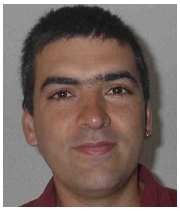
[45] ASHRAE, Measurement of Energy and Demand Savings, ASHRAE, 2002.

Please cite this article in press as: F. Terroso-Saenz, et al., An open IoT platform for the management and analysis of energy data, Future Generation Computer Systems (2017), http://dx.doi.org/10.1016/j.future.2017.08.046.

Fernando Terroso-Sáenz graduated from the University of Murcia with a degree in Computer science in 2006. He also received the master's degree in Computer Science at the same university in 2010. Since 2009, he has been working as a grant student in the Department of Information Engineering and Communications of the University of Murcia where he has published several papers in national and international conference proceedings. His research interests include complex event processing, ubiquitous computing and fuzzy modeling.

Aurora Gonzalez Vidal graduated in Mathematics from the University of Murcia in 2014. In 2015 she got a fellowship to work in the Statistical Division of the Research Support Service, where she specialized in Statistics and Data Analysis. In 2015, she started her Ph.D. studies in Computer Science, focusing her research on Data Analysis for Energy Efficiency and studied a Master in Big Data. Her research covers machine learning, data mining, and time series segmentation.

Ramallo-González completed his Ph.D. in Building Physics at the University of Exeter with a scholarship from the Wates Foundation. He has worked as post-doctoral researcher on two EPSRC funded projects in the department of Architecture and Civil Engineer of the University of Bath. Currently he is a Savedra-Fajardo Research Fellow in the Faculty of Computer Science at the University of Murcia, and PI of the project ThermaSim.



Antonio F. Gómez-Skarmeta received the MS degree in Computer Science from the University of Granada and BS (Hons.) and the Ph.D. degree in Computer Science from the University of Murcia. He is a Full Professor in the same Department and University. He has worked on different research projects at regional, national and especially at the European level in areas related to advanced services like multicast, multihoming, security and adaptive multimedia applications in IP and NGN networks.

Please cite this article in press as: F. Terroso-Saenz, et al., An open IoT platform for the management and analysis of energy data, Future Generation Computer Systems (2017), http://dx.doi.org/10.1016/j.future.2017.08.046.

Title	Applicability of Big Data Techniques to S
Authors	M. Victoria Moreno, Fernando Terroso-Sáenz, Aurora González-Vidal, Mercedes Va
Type	Journal
Journal	IEEE TRANSACTIONS ON INDUST
Impact factor (2017)	5.43
Publisher	IEEE
Volume	13
Issue	2
Pages	800 - 809
Year	2017
Month	April
ISNN	1551-3203 (Print), 1941-005
DOI	10.1109/TII.2016.26
URL	https://ieeexplore.ieee.org/abstract/
State	Published
Author's contribution	

Title	BEATS: An open IoT platform for the management and analysis of energ
Authors	Fernando Terroso-Saenz, Aurora González-Vidal, Alfonso P. Ramallo-González, Ant
Type	Journal
Journal	Future Generation Computer Systems
Impact factor (2017)	4.639
Publisher	ELSEVIER
Year	2017
ISNN	0167-739X
DOI	10.1016/j.future.2017.08.046
URL	https://www.sciencedirect.com/science/article/pii/S0167739X17304181?via%
State	Published
Author's contribution	

Appendix A

Begins an appendix