

Article

Anomaly Detection of Consumption in Hotel Units: A Case Study Comparing Isolation Forest and Variational Autoencoder Algorithms

Tomás Mendes ^{1,†} , Pedro J. S. Cardoso ^{2,*,†} , Jânio Monteiro ^{3,†}  and João Raposo ⁴ 

- ¹ Instituto Superior de Engenharia, Universidade do Algarve, 8005-139 Faro, Portugal
² Laboratory for Robotics and Engineering Systems (LARSyS) & Instituto Superior de Engenharia, Universidade do Algarve, 8005-139 Faro, Portugal
³ Instituto de Engenharia de Sistemas e Computadores: Investigação e Desenvolvimento (INESC-ID) & Instituto Superior de Engenharia, Universidade do Algarve, 8005-139 Faro, Portugal
⁴ Âmago-Energia Inteligente, 8500-581 Portimão, Portugal
* Correspondence: pcardoso@ualg.pt
† These authors contributed equally to this work.

Abstract: Buildings are responsible for a high percentage of global energy consumption, and thus, the improvement of their efficiency can positively impact not only the costs to the companies they house, but also at a global level. One way to reduce that impact is to constantly monitor the consumption levels of these buildings and to quickly act when unjustified levels are detected. Currently, a variety of sensor networks can be deployed to constantly monitor many variables associated with these buildings, including distinct types of meters, air temperature, solar radiation, etc. However, as consumption is highly dependent on occupancy and environmental variables, the identification of anomalous consumption levels is a challenging task. This study focuses on the implementation of an intelligent system, capable of performing the early detection of anomalous sequences of values in consumption time series applied to distinct hotel unit meters. The development of the system was performed in several steps, which resulted in the implementation of several modules. An initial (i) Exploratory Data Analysis (EDA) phase was made to analyze the data, including the consumption datasets of electricity, water, and gas, obtained over several years. The results of the EDA were used to implement a (ii) data correction module, capable of dealing with the transmission losses and erroneous values identified during the EDA's phase. Then, a (iii) comparative study was performed between a machine learning (ML) algorithm and a deep learning (DL) one, respectively, the isolation forest (IF) and a variational autoencoder (VAE). The study was made, taking into consideration a (iv) proposed performance metric for anomaly detection algorithms in unsupervised time series, also considering computational requirements and adaptability to different types of data. (v) The results show that the IF algorithm is a better solution for the presented problem, since it is easily adaptable to different sources of data, to different combinations of features, and has lower computational complexity. This allows its deployment without major computational requirements, high knowledge, and data history, whilst also being less prone to problems with missing data. As a global outcome, an architecture of a platform is proposed that encompasses the mentioned modules. The platform represents a running system, performing continuous detection and quickly alerting hotel managers about possible anomalous consumption levels, allowing them to take more timely measures to investigate and solve the associated causes.

Keywords: outliers detection; data quality; machine learning; deep learning; energy/water/gas anomalous consumption; buildings anomalous consumption detection platform



Citation: Mendes, T.; Cardoso, P.J.S.; Monteiro, J.; Raposo, J. Anomaly Detection of Consumption in Hotel Units: A Case Study Comparing Isolation Forest and Variational Autoencoder Algorithms. *Appl. Sci.* **2023**, *13*, 314. <https://doi.org/10.3390/app13010314>

Academic Editor: José Salvador Sánchez Garreta

Received: 23 November 2022
Revised: 15 December 2022
Accepted: 21 December 2022
Published: 27 December 2022



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The increasing impacts of climate change have caused a global awakening of individuals and governments, resulting in a necessary commitment to search for and implement

solutions capable of combating them. In Western societies, the prosperity of the adopted economic model of consumption is highly dependent on energy resources, placing high pressure on their exploitation. The global report of the International Energy Agency pointed out an increase in global energy demand by 4.6% in 2021 compared to 2020, exceeding the pre-COVID-19 pandemic levels. Global economic growth was also revised upwards and pointed to a 6.0% increase, with CO₂ emission levels rising by nearly 4.8% due to the demand for primary energy resources, such as coal, oil, and natural gas [1]. In this context, it is understood that energy consumption is inherent to economic growth and, nowadays, the consequent emission of greenhouse gases into the atmosphere somehow seems inevitable.

These results, illustrating a growing demand for energy, place serious questions on the environmental goals set for the coming years, namely the decarbonization of society by 2050. Nevertheless, the objectives are clear and the impacts are positive, involving a transition to greener society, based on three pillars: renewable electricity production, electric mobility, and energy efficiency [2].

Renewable production has been around for several years, but photovoltaic and wind production have only gained importance in the last 10 years with the technological evolution of photovoltaic panels and wind turbines. For example, in Portugal, according to data from the Portuguese Renewable Energy Association, renewable generation was responsible for supplying 65.4% of the energy consumed in 2021, avoiding the emission of 11.6 MtCO₂eq (megatonnes of Carbon Dioxide Equivalent) [3].

Electric mobility is also a reality these days, with substantial growth in the mobility market share each passing year [4]. People's acceptance of these vehicles is partly due to the development of charging infrastructures, the greater reliability of batteries, and the increase in autonomy, which is beginning to be comparable, and even in certain models superior, to vehicles powered by combustion engines [5,6].

Finally, in our list, is energy efficiency, associated with the optimization of its use, i.e., consuming less and generating more. Much has been invested into improving the energy efficiency of buildings, which represent 30–45% of global energy consumption [7] and in the integration of building management systems, i.e., systems that monitor buildings and optimize the resources for their exploration.

In the latter context, the reduction in consumption through the detection of anomalies is one area of study in which there is great interest in energy management, because anomalous consumption often represents additional losses, reflected by higher costs and environmental pollution. Several intelligent solutions have been proposed that aim to perform the early detection of defects and thus mitigate their effects [8–10], but this is not a closed issue.

The case study presented here aims specifically to describe the steps taken in the implementation of an intelligent system capable of performing the detection of non-normal consumption levels for hotel units. One of the biggest challenges in developing these solutions for hotel units is the difficult characterization of the building's consumption patterns. Sometimes, the dependence on factors such as outside temperature, solar intensity, the day of the week, the hour of the day, the building occupancy, the provision of external services (e.g., events, conferences), and unplanned occupancy, among others, cause randomness and unpredictability in the way that consumption is processed, making the task of anomaly detection algorithms very challenging [11,12]. Other problems that may arise in the development of these solutions are related to the need for a reliable data collection system or the need for a solution adaptable to different meters and hotels, creating a problem of generalization of the algorithms. To provide effective answers to the above-identified problems, general requirements were defined for the development of the intelligent system in this study, namely: (1) Integrate data from multiple sensors, which collected data at the premises of the hotel (or other) services; (2) Make use of data science methodologies to process and extract knowledge from available data; and (3) Detect anomalies associated with consumption (e.g., energy, gas, or water) in a hotel environment.

The intelligent system proposed here intends to be different from others in the sense of (a) allowing an easy adaptation to different data with distinct behaviors; (b) being compliant and easily adaptable to the different availability and combinations of variables; and (c) being computationally scalable by supporting the analysis of hundreds of data streams on a limited computational environment. In terms of (a), some appliances, such as air conditioning, are mainly used when the temperature increases above the comfort level or decreases significantly, while water heater systems require higher consumption when the temperature is low (possibly being part of the year disconnected, when supported on solar water heaters). Another example is water consumption, which has variations but is always present, with peaks dependent on the season and hour of the day. In terms of (b), distinct hotels have different sets of meters, information concerning occupancy, or meteorological data. Thus, the implemented system must be capable of adapting to whatever variables are available. Finally, related to (c), the system is expected to make predictions (anomaly detection) but also ML/DL model updates (fitting of new data) while supported in limited computer resources and multiple data streams.

Thus, as a study case, the work can be summarized as follows. (i) Data from a hotel were used as the main driver for the study of two anomaly detection algorithms. (ii) As can be expected from real scenarios, the developed EDA revealed data problems (e.g., readings are missing and erratic reading peaks) which led to the investigation and adoption of methods to repair those data, before it was presented to the anomaly detection algorithms. At this level, the interquartile range was used to detect possible outliers and a linear weighted moving average was used to repair missing data. (iii) A metric to measure the accuracy and tardiness of the anomaly detection algorithms was proposed. (iv) A comparison between the isolation forest algorithm and a variational autoencoder was made, allowing us to validate that, for a system with limited computational resources and expected to look for anomalies in many meters of different hotels, the former is probably the best solution as an anomaly detection method. Finally, (v) the architecture for an anomaly detection intelligent system is proposed.

With the system requirements established, the study was organized in the following way. A brief state-of-the-art analysis of anomaly detection in buildings consumption is presented in Section 2. The section starts by introducing the anomaly detection problem and then analyzes some developments and case studies in the building consumption anomaly detection field. Then, in Section 3, the theoretical background behind the algorithms studied here is explained and a newly proposed metric to compare the performances of the algorithms in unsupervised time series is presented. Section 4 provides an exploratory data analysis, introducing the procedures performed to analyze the data and the methods used to correct the identified readings problems. Section 5 describes the solutions that support the module for data treatment. Section 6 presents the comparison study between the IF and VAE algorithms. Finally, the proposal of a platform, the conclusions, and some future steps are presented in the last two sections.

2. Anomaly Detection in Building's Consumption: A Brief Overview of the State of the Art

Nowadays, the development of intelligent platforms hardly exists without the use of ML algorithms [13,14]. Before proceeding to describe the methodology used in the detection of anomalous consumption levels (Section 3), this section presents an introduction to the anomaly detection problem and an analysis of the state of the art at the algorithmic level.

An anomaly detection problem can be defined by finding points or patterns in the data that are very distinct from the rest [15]. Anomaly detection is an area of broad scope that can be used in many fields of knowledge, such as fraud detection [16–18], image processing [19], healthcare [20,21], equipment failure detection [22,23], or consumption [9,24]. Anomaly detection problems are generally divided into three main categories [15,25]: (i) Supervised; (ii) Semi-supervised; or (iii) Unsupervised detection. Supervised detection is characterized by problems wherein data are classified, i.e., anomalies and normal data are previously

known. Detection approaches are usually based on algorithms that compare the new data with the classified one [26]. Semi-supervised detection is considered when only data regarding the normality of the system are known. Typically, detection algorithms train with these data and are tested with unseen datasets in order to assess the presence of anomalies [27,28]. Finally, unsupervised detection, unlike supervised and semi-supervised solutions, is supported on unlabeled data. These algorithms usually assume that few anomalies exist in the dataset and they have different characteristics from the rest [29,30].

Bearing in mind the challenge of the detection of consumption anomalies, it was understood that we should fit our anomaly detection problem in the category of unsupervised detection algorithms, since the data available for conducting the study were unclassified, as will later be seen in Section 4. Furthermore, the system requirements demand a solution applicable to different hotels and meters, with the possibility of dynamically adding new ones to be analyzed on the fly.

As already mentioned, different algorithms exist for different types of anomaly detection problems. For example, Xiong et al. [31] proposed a data anomaly detection method that combines an IF and an AE algorithm. The IF algorithm computes the anomaly score of energy data, and then the lower anomaly score data are chosen for model training. Then, new data pass through the AE, trained with “normal” data, and records whose reconstruction error was larger than a predefined value are identified as anomalies. Experimental results of the combined method, on the electricity consumption dataset, achieved an F1 score of 0.981. Kaymakci et al. [32] presented an end-to-end solution of an anomaly detection system. The system uses the concept of a Long Short-Term Memory-based AE as an unsupervised learning model. Himeur et al. [33] proposed an anomalous energy consumption detection system at the appliance level, using an AE and micro-moments. In their work, the energy usage history of household appliances and occupancy patterns are used for the building of a normal energy consumption behavior model. Energy micro-moments are fixed when consumers turn an appliance on or off, change the energy consumption of a specific appliance, or enter (or leave) a given site. The energy micro-moments are associated with users’ daily tasks and their patterns are fed into an AE model. The model included the detection of two kinds of anomalous energy consumption, namely: “excessive consumption” and “consumption while outside”.

Apostol et al. [34] presented a rule-based decision system for anomaly detection in multivariate time series using change point detection. Their solution automatically manages to identify anomalies and remove the false positives originated by change points. They also propose a metric based on the record to be an anomaly or a change point. The experiments use a dataset with multivariate time series about water consumption. Liu et al. [35] proposed a data mining-based framework that extracts electricity load patterns and discovers hidden information on them. A clustering method was also proposed to identify the typical electricity load patterns. The proposed framework was applied to analyze the electricity consumption data of office buildings. Lipčák et al. [36] presented a big data platform for the detection of anomalies in power consumption. They demonstrate the application of the system to a scheme of power consumption anomaly detection, benchmarking different alternatives. For a whole building, Chiosa et al. [37] proposed an anomaly detection and diagnosis methodology to detect meter-level anomalous energy consumption and then perform a diagnosis on the loads responsible for anomalous patterns. The process consists of several steps combining data analytics procedures. The methodology was developed and tested on the monitored data of a medium/low voltage transformation unit of a university campus.

Some other works can be found that perform surveys on several methods. For instance, the work by Al-amri et al. [38] presented a review of ML and DL techniques for anomaly detection in IoT data. The work includes an analysis of the nature of the data, anomaly types, learning modes, window models, datasets, and evaluation criteria. Furthermore, research challenges related to data evolution, feature-evolving, windowing, ensemble approaches, nature of input data, data complexity and noise, parameter selection, data

visualizations, heterogeneity of data, accuracy, and large-scale and high-dimensional data were studied. Furthermore, for the energy consumption of buildings, a review of anomaly detection systems based on artificial intelligence was presented by Himeur et al. [24]. Their work includes a taxonomy to classify existing algorithms based on different modules and parameters, such as ML algorithms, computational platforms, anomaly detection degrees, feature extraction procedures, and usage scenarios. Domain-specific issues, difficulties, and challenges that remain open are also analyzed. Those challenges include the deficiency of having definitions of anomalous power consumption, annotated datasets for supervised and semi-supervised learning, standard metrics to evaluate the models' performance, and the reproducibility of scenarios, and privacy preservation.

In general, the studies are made with single sources of data, and most use well-known datasets. The controlled quality datasets are used to evaluate the performance of the models, providing different conditions than the ones that are proposed in this case study. In our case, we are looking for a semi-automated process, adaptable to different sources of data and combinations, to be implemented without major tuning, and without major knowledge by the deployers. In this context, the methodology for this work is presented next, including the two unsupervised algorithms which were analyzed (namely the IF/ML approach and the VAE/DL approach) and a new performance metric.

3. Methodology

The methodology adopted for the development of the study follows a classical ML flow (Figure 1). The first part was the collection of the data, which were obtained from meters located in hotels in the region of Algarve, Portugal. Then, an EDA was developed, revealing some of the data problems, such as missing and erratic-peaks (Section 4). Some of the revealed problems were solved by the implementation of a data correction module (Section 5). With the initial data analysis transformation applied to the data, some algorithms for anomaly detection were considered. This selection was supported in works such as those ones from Pereira and Silveira [39], Zhou and Paffenroth [40], Chen et al. [41], Yao et al. [42], and Cook et al. [13] (Section 3.1). Then, to evaluate the models, a metric to measure the accuracy and promptness/tardiness of the anomaly detection algorithms was then proposed (Section 3.2). Finally, the tested algorithms were evaluated (Section 6) and an architecture for an anomaly detection intelligent system is proposed (Section 7).

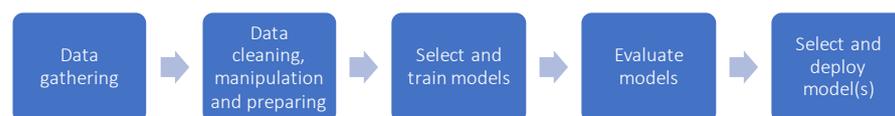


Figure 1. Flow diagram of the adopted methodology.

3.1. Algorithmic Approach

This section details the algorithms that were later used for the model selection. Two algorithms were considered, namely: isolation forest and variational autoencoder.

3.1.1. Isolation Forest

In the field of ML, the isolation forest [43] algorithm belongs to the unsupervised machine learning group of algorithms. This method works a bit differently from the typical methods of clustering, which are consistently associated with unsupervised detection. Furthermore, IF was used in the past to handle anomaly detection problems that involve time series, similar to those evaluated herein (e.g., in [44,45]).

This method consists of a set of elements called isolation trees. Isolation trees are built by isolating the points of the given dataset, $X = \{X_1, X_2, \dots, X_j\}$, through a recursive division of the sample space. In this recursive division, a feature belonging to the elements of the sample space (X) is randomly chosen. Then, also randomly, a division value (p) is computed within the range of values of that same feature, as shown in the example depicted in Figure 2. Supported by the fact that theory tells us that anomalous observations

are usually different and exist in small numbers, it is expected that those anomalous observations would demand a smaller number of divisions to be isolated, because they should be more distant from the other ones. In the end, the process is equivalent to the logic of a collection of trees, i.e., a forest is obtained.

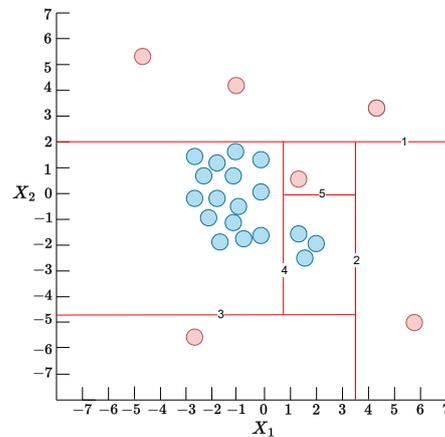


Figure 2. An example of the process that occurs in an isolation tree building to isolate points in a two-dimensional sample space.

The classification of samples as anomalies or normal is supported in a scoring system given by the equation

$$s(x, n) = 2^{-\frac{E(h(x))}{c(n)}},$$

where for a given x , $h(x)$ corresponds to the length of the path from the root node to the outer node that isolates x in a given forest tree, $E(h(x))$ is the average length of these paths in the forest, and $c(n)$ is the average length of the isolation trees for n samples (n is the number of samples in the set X). This value is computed by equation

$$c(n) = 2H(n-1) - (2(n-1)/n),$$

where H is a harmonic number [43].

The rationale of the IF algorithm is to use multiple trees (as sketched in Figure 3) to increase the effectiveness of anomaly detection. As the splitting process is made independently in different trees, the randomness that results from multiple decision trees is lower than that obtained from a single tree.

The algorithm has been used to solve several anomaly detection problems. For example, Puggini and McLoone [46] compared four methods for variable reduction and used the IF algorithm in anomaly detection, based on data from the semiconductor industry. To understand which approach was better for the study, they carried out two case studies, from which they concluded that there would be some advantages in using certain reduction methods over others. They also proposed an anomaly detection system based on the IF method. Ding and Fei [45] also presented a general architecture for anomaly detection in streaming data with the IF method, and suggest an approach where they associate the IF with a sliding window method, a well-known approach for time series. To evaluate the method, they performed several tests using four datasets, which allowed them to validate the effectiveness of the algorithm. Another interesting work, carried out by de Santis and Costa [47], consisted in comparing different algorithms in the identification of faults in small hydroelectric power plants. From the results obtained, they concluded that the IF algorithm and its variant, the extended isolation forest, were the algorithms that stood out the most, having good characteristics to be adopted in an online detection system. The authors concluded that these are also solutions that require a low computational cost.

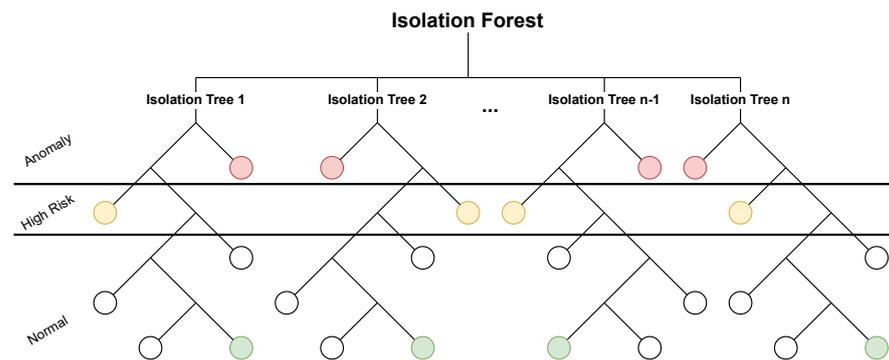


Figure 3. Sketch of anomaly detection with IF.

3.1.2. Variational Autoencoder

The variational autoencoder, like other autoencoders (AEs), is composed of two parts: (i) the encoder, which performs the learning/compression of the data into a reduced dimension, called latent space, while preserving the most relevant information; and (ii) the decoder, which performs the reconstruction of the input data with the smallest possible loss [48,49]. Figure 4 illustrates the architecture of AEs, as defined above: the encoder on the left side, the compressed information on the middle, and the decoder on the right side.

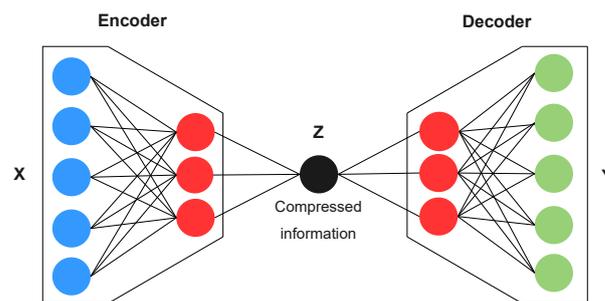


Figure 4. Sketch of the AE base architecture.

The main characteristic of the VAE is associated with the modeling of the latent space, Z , as a function of a random variable with a given type of distribution. Usually, an isotropic normal distribution is applied, because it is considered that the relations between variables in a reduced dimension will be less complex than in the input dimension.

The architecture of the VAE used in this study is represented in Figure 5. In this context, the work of Pereira and Silveira [39] was used as a baseline for the VAE design and associated parameterization, including the number of neurons, activation functions, and layers used. The encoder (an orange element in the figure) and the decoder (green element in the figure) were both designed with a bi-directional layer of 128 Long Short-Term Memory (LSTM) cells in each direction [50,51], making a total of 256 cells, with \tanh activation function, and as result, the concatenation of the two layers. The latent space Z was modeled according to an isotropic normal distribution [52,53], with the parameters μ_z and σ_z being obtained through two fully connected artificial neural network layers, which connect to the last state (h_t) of the encoder, with linear activation and SoftPlus, respectively. The latent space Z is calculated using the approximation $Z = \mu_z + \sigma_z \epsilon$, where ϵ is a random value of the chosen distribution that acts as an auxiliary noise. The latent space vector, Z , is repeated t times and then, in the decoder, data are reconstructed into the original form. The detection of anomalies with this model can be carried out based on the reconstruction error or, as it is a probabilistic model in which the parameters of a distribution are reconstructed, it is possible to use probabilistic measures for the classification of anomalies [53]. In our

case, we opted for a more conventional solution, i.e., detection by reconstruction error and the use of a sliding window to perform the identification of potential anomalies.

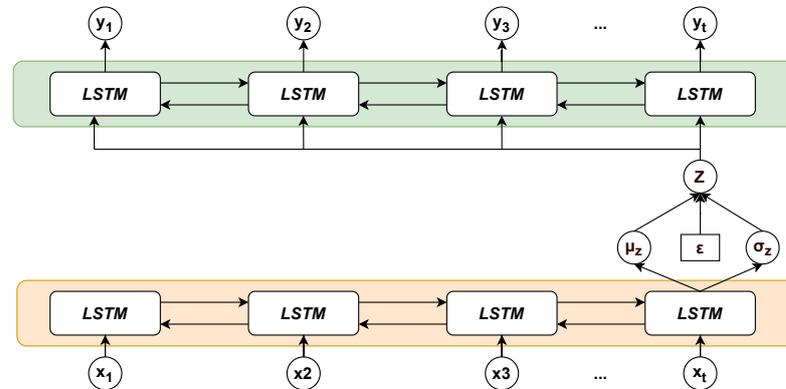


Figure 5. Architecture of the used VAE. The encoder is represented in orange and the decoder in green, with bi-directional layers of Long Short-Term Memory cells in each direction and the latent space Z modeled according to an isotropic normal distribution (with parameters μ_z and σ_z).

There are several recent works where the VAE algorithm has been used for unsupervised anomaly detection. The one made by Pereira and Silveira [39] stands out, where an VAE was applied to the anomaly detection on photovoltaic production data, in order to identify problems with possible causes on the weather or inverters failure, for example. In another work, Xu et al. [54] proposed a model named “Donut”, a VAE-based approach to perform unsupervised anomaly detection for seasonal KPIs in web services. The author concluded that the “Donut” model outperformed the state of the art at that time. Furthermore, Yao et al. [42] made a comparison study of algorithms in different datasets to identify anomalies, with VAE-based feature extraction as one of the approaches.

3.2. Performance Metric

Performance metrics are fundamental for evaluating algorithms’ performance. In the case of supervised learning, common metrics for classification problems, such as confusion matrices, accuracy, F1-score, precision, or recall, were used in some works in the field (e.g., in [31,38]). However, while treating unsupervised time series or measuring the promptness of detection on streaming data, those metrics do not apply. The Numenta Anomaly Benchmark (NAB) [55] shows a way to evaluate the performance of algorithms with streaming data. To do that, the algorithms are tested in the labeled datasets available to the effect and a group of metrics evaluates the performance of the algorithms in many topics, like if all anomalies were detected or the promptness of detection. Here, it was decided to propose a metric, founded in the NAB principles but more adapted to our unsupervised problem concerns, as we will see below.

The proposed performance metric is formulated by combining 2 scores: S_1 , which is the score that indirectly evaluates the incidence of supposed false positives, and S_2 , which is the score referring to the promptness classification of anomalies. The final score, S_{final} , combines the previous two and is responsible for assigning the algorithm’s performance score for a given test scenario. The score values are in the interval $[0, 1]$, with 1 as the best performance.

In more detail, S_1 is obtained using the equation

$$S_1 = \frac{1}{1 + \exp\left(\frac{n_{va} - n_{up} \times k_1}{n_{up} \times k_2}\right)}, \tag{1}$$

where n_{va} translates a number of unknown points detected as an anomaly by the algorithm (i.e., that do not belong to the anomalies that were introduced), n_{up} is the total number of unclassified points in the test sample (number of points in test sample minus

the number of anomalies introduced), and k_1 and k_2 are the values that allow adjusting the associated penalty level. With the score S_1 , it is possible to evaluate the algorithm in terms of false positives giving tolerance to the classification of uncertain points as anomalies. In our case, $k_1 = 0.1$ and $k_2 = 0.01$ were considered. In this case, when the algorithm detects up to 5.0% of false positive anomalies, a small decrease is introduced on the value of S_1 . When 10% of the readings are marked as false positives, then S_1 would be equal to 0.5.

As an example, Figure 6 graphically represents the evolution of the S_1 score as a function of the number of unknown points detected as an anomaly (n_{va}) for a total number of unclassified points (n_{up}) of 8112. This was the scenario studied during the performed tests (Section 6). For example, if the algorithm classifies 300 readings as anomalies, beyond the anomalies introduced (i.e., $n_{va} = 300$), in a universe of $n_{up} = 8112$ samples, the performance of the algorithm will not be badly evaluated, since 5% of the 8112 is 405.6. However, if the value of n_{va} increases, a penalty will begin to exist which is reflected in the S_1 score. For instance, with $n_{va} = 750$, the value of S_1 would decrease to 0.68. Thus, this sigmoidal shape, adjustable by the formula's parameters (n_{up} , k_1 , and k_2), allows for being more or less flexible in the identification of anomalies that were not identified as such in the dataset.

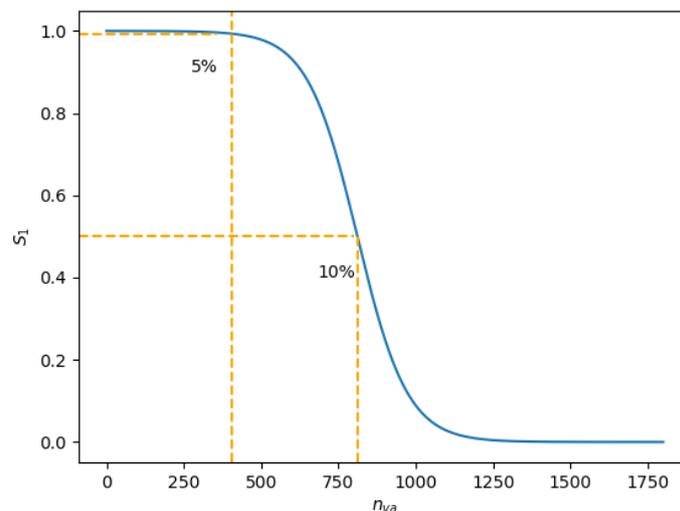


Figure 6. Illustrative example of the evolution of the S_1 metric as a function of the number of unknown points classified as abnormal (n_{va}), with $n_{up} = 8112$.

The score S_2 is obtained for each interval of anomalies from equation

$$S_2 = \frac{1 - \exp\left(k_3 \frac{R_f - A_t}{A_t - A_s}\right)}{1 + \exp\left(k_3 \frac{R_f - A_t}{A_t - A_s}\right)} \quad (2)$$

where the indices A_s and A_t translate, respectively, the first and last index of the anomaly window, R_f is the first reading detected inside the window, and k_3 is a parameter that was set to 10 in this case study. S_2 scores the performance in classifying the anomalies, the speed of detection, and false negatives. The equation was adjusted to start the penalty depending on the extension of the anomaly and to decrease more sharply in the middle of it. The score S_2 is equal to 0 when the anomaly is not detected, reflecting the non-detection of a true anomaly.

Figure 7 shows the evolution of S_2 as a function of R_f , for an anomaly consisting of 24 points, which, for the tests carried out, would correspond to an anomaly lasting 24 h. When the R_f is less than 12, that is, the anomaly detection is performed in the first 12 h, there is a small penalty, and only when the detection delay goes beyond that threshold

does the algorithm's performance start to be penalized. Parameter k_3 allows for more or less flexible in the promptness/tardiness in identifying known sets of anomalies.

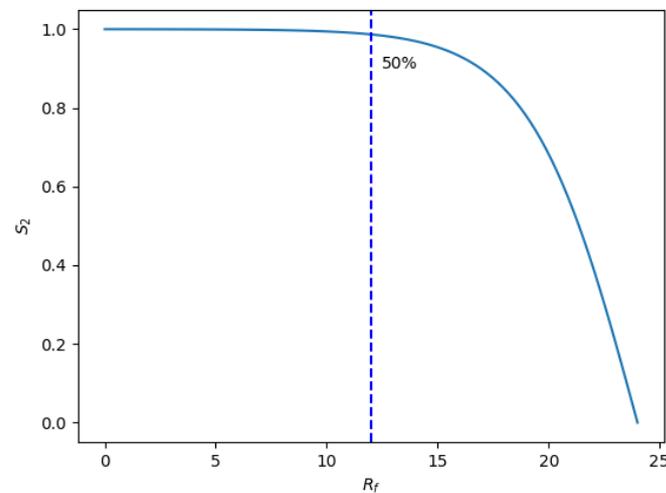


Figure 7. An illustrative example of the evolution of the value of S_2 as a function of the value of R_f , for an anomaly of a sequence of 24 readings.

Finally, the value of the algorithm's performance metric is given by

$$S_{final} = S_1 \frac{\sum_{i=1}^n S_{2,i}}{n}, \quad (3)$$

where $S_{2,i}$ is the values of S_2 for the anomalous range i . The equation computes the average of the anomaly detection scores (S_2), as different anomalies with different extensions might exist, penalized by the classification of unknown points as anomalies (S_1). In the end, the intention of the final metric value, S_{final} , is to reflect a balance between the detection of anomalies and the incidence of false positives. For example, $S_{final} \approx 1$ means that all anomalies were detected quickly and the number of points from which the algorithm classification could not be assessed remained low.

4. Exploratory Data Analysis

Before performing the tests with ML algorithms, an initial study of the data was carried out, that is, an EDA phase was performed. The objective of the EDA was to understand the nature and characteristics of the data, identifying possible problems that in the future may impair the performance of ML models [56,57].

The initial approach of the EDA phase was performed using graphical representations and a statistical analysis of the consumption. In terms of statistical analysis, a simple observation of the mean, median, and quartiles was enough to find possible anomalies, in some cases not related to the consumption itself.

The datasets used in the EDA refer to one water meter, a gas meter, and two electricity meters from a hotel unit in the Algarve, Portugal. At this stage, and given the diversity of the analyzed cases, the data were considered to be representative of the types of time series that will be analyzed by the system in the future. We should notice that readings were sampled hourly and refer to the period between January 2014 and October 2021. Furthermore, in this context, a “problematic observation” is simply a purely bad quality observation caused by technical issues while an “anomaly reading” should be an effective problem with consumption.

4.1. Graphical Representation of the Time Series

The first EDA approach consisted of the graphical representation of the different time series, shown in Figure 8. For the analyzed meters, one of the identified problems was the

existence of occasional peaks in readings. It was later understood that these values were associated with data transmission failures by the hotel's centralized technical management system (CTMS). When this failure occurs, the readings are accumulated on the sender side until the transmission is resumed. At this point, the accumulated value is transmitted, producing both: (1) an apparent peak in consumption, which effectively did not occur; and (2) an interval of time without readings, as shown with more definition in Figure 9. These peaks and gaps in the data, due to transmission errors or lack, should not be confused with anomalous consumption.

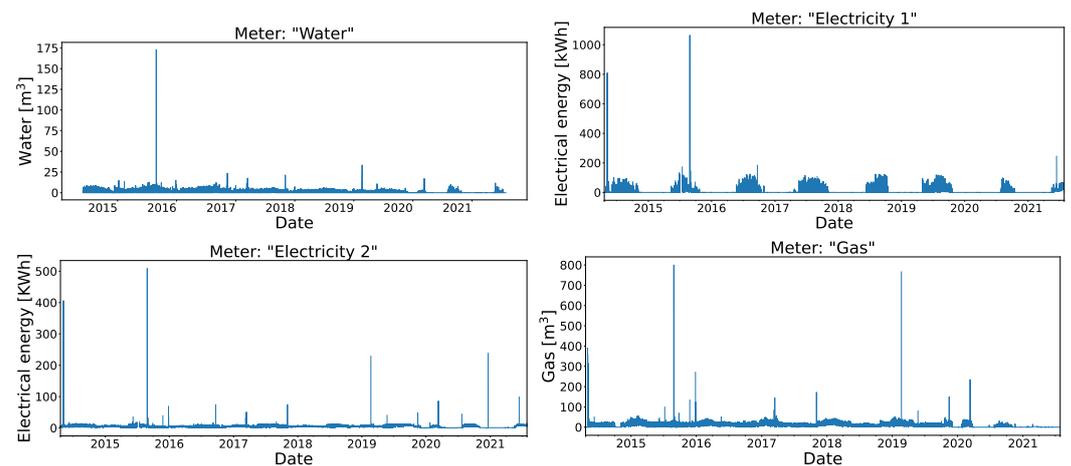


Figure 8. Representation of the readings series (sampled hourly), associated with different hotel meters, where occasional consumption peaks are clearly visible.

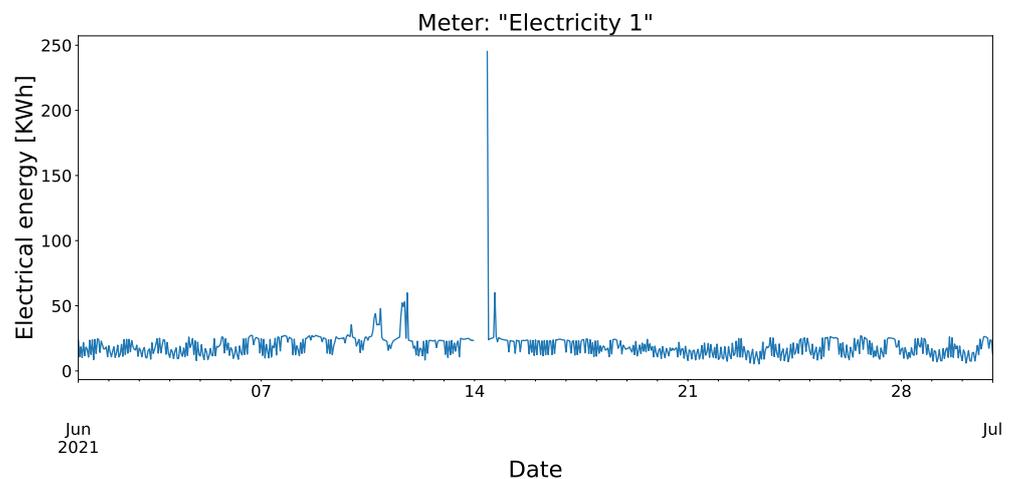


Figure 9. Detail of technical failure in readings (hourly sampled) received by meter “Electricity 1”.

Other analyses were also carried out on the data, to understand its behavior over time. For example, Figure 10 shows the weekly consumption in distinct years for the “Electricity 1” and “Gas” meters, showing that the consumption tends to follow a pattern, due to the annual seasonality of tourism in that hotel. The year 2020 looks a little out of the ordinary due to the COVID-19 pandemic and the year 2021 is not represented in this analysis because the available readings were not complete. From a daily perspective, analyzing a few random weeks of the different meters, a periodicity was also verified on different days of the week, concluding that, typically, from Monday to Sunday, consumption tends to maintain an hourly pattern, as shown in Figure 11.

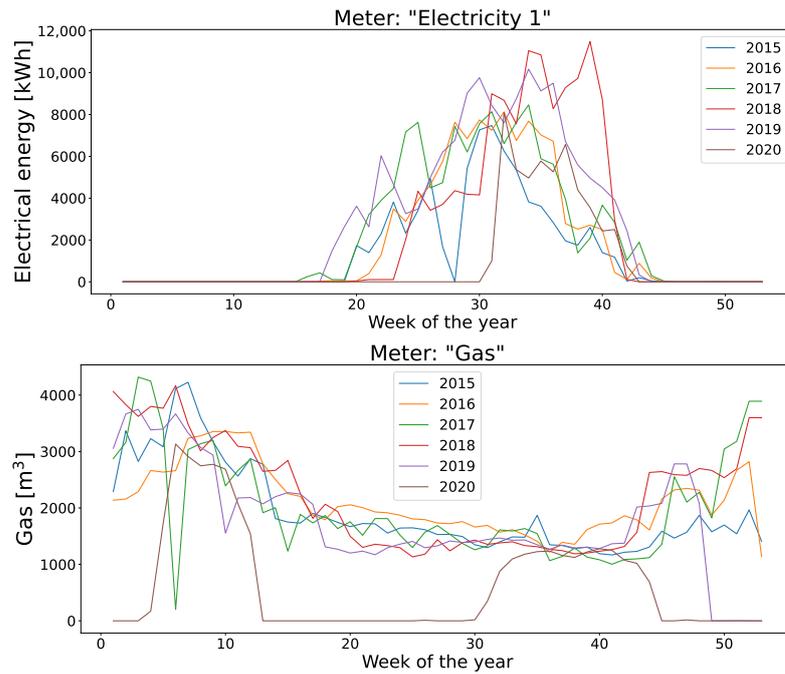


Figure 10. Yearly analysis with weekly sampled readings for meters "Electricity 1" and "Gas".

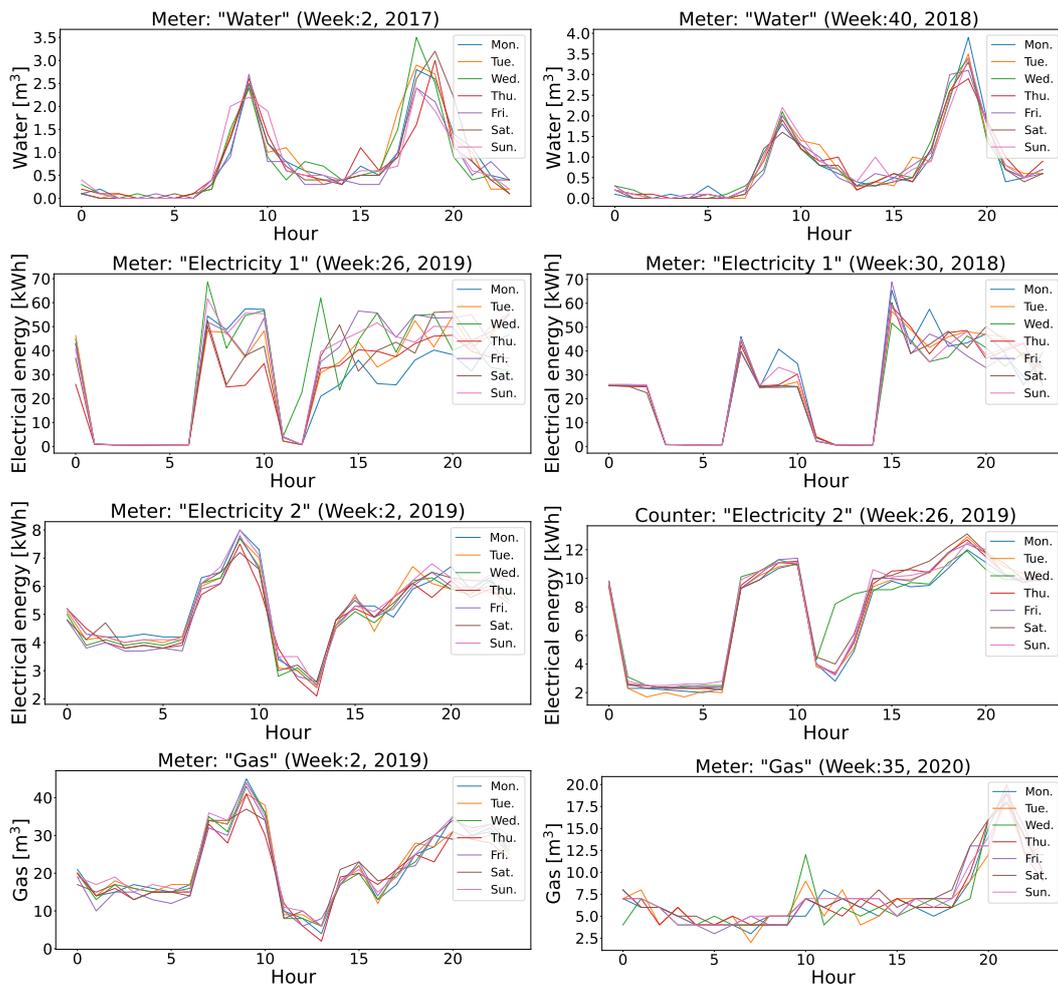


Figure 11. Representations of daily consumption in different days of a week, for the 4 m under study.

4.2. Statistical Analysis

In a second phase, a statistical analysis of the 4 m was carried out with the computation of the mean, standard deviation, quartiles (Q_1 , Q_2 e Q_3), and the minimum and maximum values of the readings, as presented in Table 1. From these data, some conclusions could be drawn. The maximum value identified in the four meters is very far from the values recorded in the third quartile (Q_3). Specifically in the case of the “Electricity 1” meter, there is at least one reading with 1064 kWh of hourly electrical consumption, with a similar situation being verified for the “Water” meter, with an hourly reading of 173.10 cubic meters (m^3). These situations clearly suggest the possibility of problematic or anomalous data in the available readings. Another value that suggests the existence of errors in the data is the value of the standard deviation, showing a large range of variation in some readings in relation to the mean value, which again suggests the presence of problematic readings.

Table 1. Statistical description of the meters’ readings in the analysis.

Meters	Water (m^3)	Electricity 1 (kWh)	Electricity 2 (kWh)	Gas (m^3)
mean	0.83	4.73	11.61	9.95
std	1.39	4.13	21.89	10.66
min	0.00	0.00	0.00	0.00
25%	0.00	2.70	0.00	4.00
50%	0.40	4.30	0.00	9.00
75%	1.10	6.90	18.80	14.00
max	173.10	508.60	1064.60	800.00

4.3. Histograms

Histograms were used in the last phase of the analysis, which have shown to be important to confirm the results obtained in the last steps. As can be seen in Figure 12 (note that the scale of the ordinates is logarithmic), it is possible to perceive which ones are the most common intervals of values and to anticipate whether the data obey some known distribution or not. The readings previously highlighted in the statistical analysis as possible anomalies are readings with unitary frequency, which, attached to their distance from the others ones and knowing the physics associated with the respective meters, (almost) guarantees that they will be problematic.

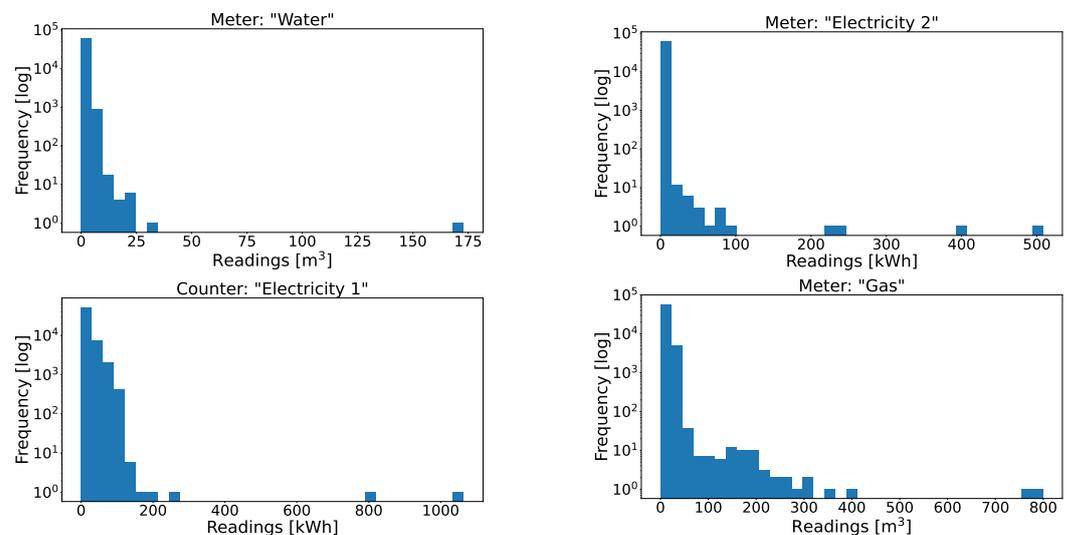


Figure 12. Histograms of the “Water”, “Electricity 1”, “Electricity 2” and “Gas” meters with the readings from January 2014 to October 2021.

4.4. Conclusions of the EDA

With the EDA, which was summarized in this section, it was concluded that data are affected by seasonality (e.g., annual and weekly) and that the biggest problems affecting data quality are: (i) outliers/unreliable and (ii) missing readings. The origin of the problems noticed at this stage was later justified by technical issues that resulted from transmission errors or gaps. This is in fact a problem that affects many sensor networks and should not be confused with consumption anomalies. To solve the mentioned problems, statistical approaches were chosen to automatically identify them and then introduce estimated values to correct anomalous ones (or fill the gaps in the readings), resulting in the data correction module. This was thought to be necessary to guarantee a better historic registry of readings to our promoter, and data with higher quality/reliability will be used by the ML alarmistic model, as will be explained next.

5. Data Correction Module

The analysis of the datasets performed in the EDA phase has shown that the anomaly detection system must integrate a data correction module. In fact, this module should be capable of detecting faulty transmissions before those data reach the machine learning module. The development made in this work has led to the specification, and the later implementation, of a solution that should be capable of: (i) detecting and removing problematic observations; and (ii) introducing missing values when a period of data loss is identified. In the following, we describe these features of the data correction module.

5.1. Detection and Removal of Problematic Observations

Problematic observations are values that directly interfere with the quality of the data and that could generate false alarms or later reduce the identification of real anomalies, if the training of the algorithms is performed with these values. Two statistical methods were considered to perform these functions: the interquartile range (IQR) [58,59] and standard deviation method (SDM) [56,60].

The use of the SDM was the first to be considered. However, the analysis carried out on data (Section 4) proved that it might not be sufficiently robust, because it is not guaranteed that meters conform to a normal distribution, as this method requires. For that reason, the adopted solution was the IQR method, as our data tend to be skewed and have fewer limitations associated with the distribution of data. The IQR method is very simple to apply as it considers values in the range $[Q_1 - K \times IIQ, Q_3 + K \times IIQ]$ to be “normal”, where Q_i is the i^{th} quartile, K is an adjustment parameter (usually defined as 1.5), and $IIQ = Q_3 - Q_1$ is the interquartile range.

As an example of this data treatment, Figure 13 presents the resultant data after removing the peak values shown in Figure 8. The data quality has significantly improved and virtually all readings that could be visually considered problematic have been removed. In this context, it was found that this IQR method must be adjusted by setting the tuning parameter K according to each meter, in order to preserve the maximum number of correct readings. After carrying out this procedure, the missing values were estimated and reinserted, as described in the following section.

5.2. Missing Values Imputation

The imputation of missing values in time series can be essential to ensure its viable use in certain ML algorithms, such as the ones supported in LSTM cells, which depend on the data sequence. For this reason, and in order to support a detailed record of readings with better quality and reliability, a solution was required that would allow estimating missing values due to, for example, transmission failures.

The use of more complex solutions based on ML algorithms to perform this task would add a whole complexity to the system. For this reason, the method chosen was the linear weighted moving average (LWMA) algorithm, formulated according to equation

$$R_{weighted} = \sum_{i=1}^n R_i w_i, \tag{4}$$

where $R_{weighted}$ is the weighted reading, R_i ($i = 1, 2, \dots, n$) are readings from the n previous days that occurred at the same time of the reading being weighted, and each past reading is affected by a weight $w_i = \frac{2(n-i+1)}{n(n+1)}$ ($i = 1, 2, \dots, n$).

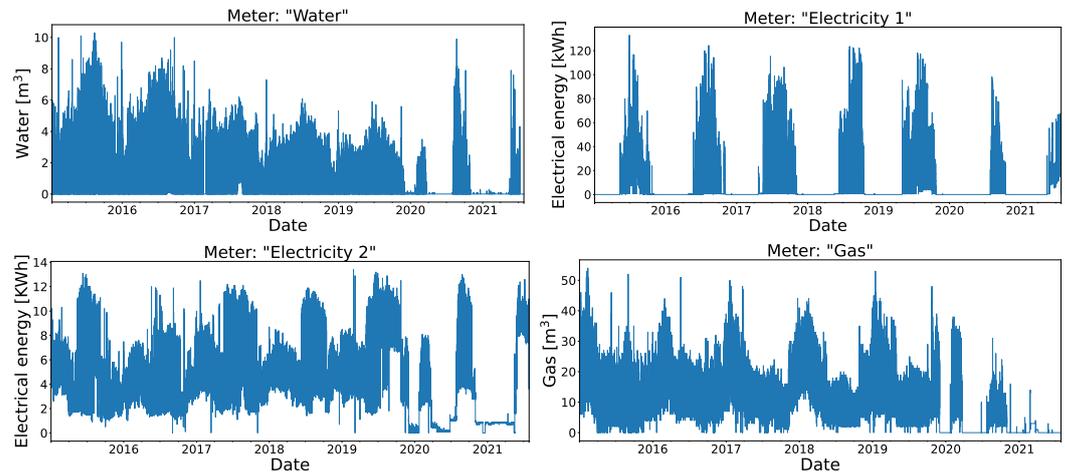


Figure 13. Representation of corrected time series, associated with the different hotel meters.

This solution is supported by the fact that most meters have a weekly frequency (see Section 4.1) and, as can be seen in the Tables 2 and 3, there is a bigger correlation between the most recent readings in the different analyzed meters. The LWMA algorithm has been used in other works where it was necessary to impute values, in which the authors concluded that the algorithm was effective in performing this task [61,62].

Table 2. Values of existing correlations between readings at 12:00 A.M. (midday) and readings that occurred at the same time on previous days.

Number of Days before	Water	Electricity 1	Electricity 2	Gas
1	0.66	0.86	0.91	0.86
2	0.65	0.83	0.88	0.83
3	0.63	0.82	0.86	0.81
4	0.62	0.79	0.84	0.80
5	0.61	0.78	0.83	0.79

Figure 14 shows the results obtained using the LWMA method for the introduction of the missing values and filtered readings, in the “Electricity 1” meter. As can be verified, the time series was reestablished and the new values are “imperceptible”, making this approach a reasonable solution to this problem.

The realization of tests with ML and DL algorithms needs data integrity and quality to produce relevant results. With the data corrected using the described solution, in the following section, a presentation is made of the tests, the results achieved with the algorithms, and the conclusions obtained from them.

Table 3. Values of existing correlations between readings at 12:00 P.M. (midnight) and readings that occurred at the same time on previous days.

Number of Days before	Water	Electricity 1	Electricity 2	Gas
1	0.63	0.96	0.96	0.92
2	0.59	0.95	0.95	0.90
3	0.55	0.93	0.93	0.89
4	0.52	0.91	0.91	0.88
5	0.50	0.88	0.90	0.87

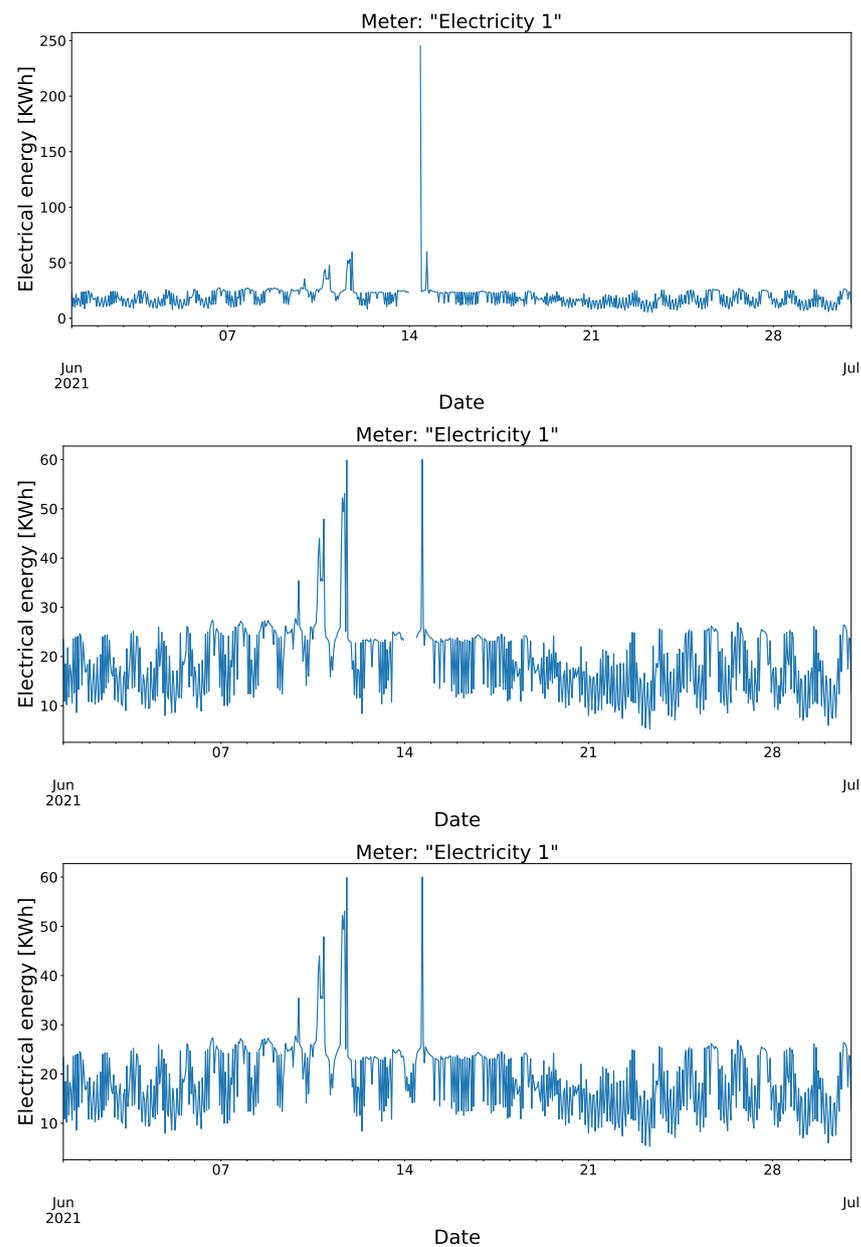


Figure 14. Pre-processing of the data collected by meter “Electricity 1”: original data (**top**), after removal of peaks (**middle**), and after the input of estimated values for the missing values (**bottom**).

6. Comparison between Isolation Forest and Variational Autoencoder Algorithms

This section compares the performance of the IF and VAE algorithms on the classification of anomalies in an unsupervised time series of hotel consumption. The comparison

was made using a battery of tests developed by the authors, which were evaluated using a newly proposed performance metric and based on the computational costs that each method imposes.

6.1. Established Dataset

The established dataset consists of 2 years (2015 and 2016) of hourly measurements to perform the training (17,520 readings), followed by 1 year (2017) of data for the tests (8760 readings), considering the independent analysis for the four meters in this study. The considered period was chosen according to the results of the EDA and the convergence between the variables' registry, i.e., the availability of other features used in the algorithms (e.g., hotel occupancy), as we will see later. Furthermore, following a strategy reviewed by Al-amri et al. [38], the sequence of anomalies was explicitly forced in the test year to simulate excessive consumptions (three periods of 24 readings and one long period of 576 readings), as depicted in Figure 15 (marked in red). The different scenarios tested (which included a combination of the data with the parametrization of the algorithms), which will be explained next, had the following main objectives: (i) evaluate the performance of the algorithms in the classification of anomalies in different meters; (ii) discover a possible set of variables and hyper-parameters to be used as a baseline in the training of models; and (iii) study the behavior of the algorithms when exposed to different combinations of variables, since the detection system needs to be very adaptable to this type of situation.

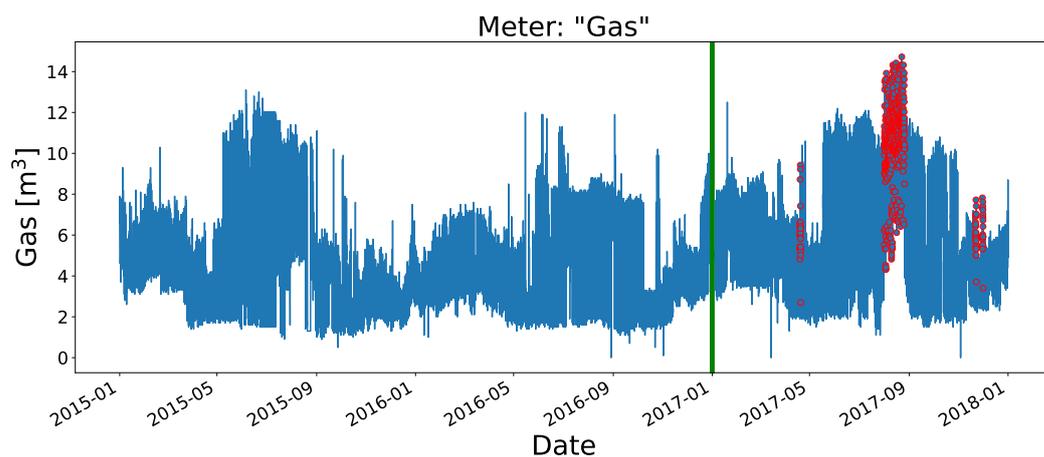


Figure 15. Graphical representation of the dataset of one meter used in the computational tests. The green line separates the train and test periods; the anomalies introduced in the test year are in red.

As already mentioned, to understand the level of precision in the detection of outliers by the algorithms in the study, four anomalies were introduced into the original test dataset, giving us a more controlled environment.

6.2. Features Scenarios

Four scenarios of input variables were considered, as shown in Table 4. These scenarios include consumption features, temporal variables, hotel, and environment variables. Those sets of features give more “context” to the readings and should improve the results. Some of the variables based on meter readings were created in the process of data engineering, such as:

Δ_X —the difference between the current reading and the reading obtained X hours before, e.g., Δ_{2h} is the difference between the current reading and the reading that occurred 2 h earlier. X assumes six values: 1 h, 2 h, 3 h, 24 h, 48 h, and 72 h;

$\Delta_{\text{average}_{24h}}$ —the difference between the current reading and the average of the last 24 h;

\min_{24h} —identifies the minimum consumption of the last 24 h.

Binarization of these parameters was also tested, identified with a super-scripted b in features detailed in Table 4, whose objective was to transform the differences into 0 or 1, depending on whether the consumption increased or decreased. Temporal variables and variables associated with the hotel were also used, such as, e.g., “temperatures”, “occupancy”, and “degree day” (measurement unit for the building’s heating/cooling needs), as suggested by Eras et al. [11].

Table 4. Different sets of input variables of the algorithms used in the tests.

Scenario	Features	Temporal Variables	Hotel and Environment Variables
1	Reading, Δ_{1h} , Δ_{2h} , Δ_{3h} , Δ_{24h} , Δ_{48h} , Δ_{72h} , \min_{24h} , $\Delta_{\text{mean}_{24h}}$	Hour, day of the week, month	None
2	Reading, Δ_{1h}^b , Δ_{2h}^b , Δ_{3h}^b , Δ_{24h}^b , Δ_{48h}^b , Δ_{72h}^b , \min_{24h} , $\Delta_{\text{mean}_{24h}^b}$	Hour, day of the week, month	None
3	Reading, Δ_{1h} , Δ_{2h} , Δ_{3h} , Δ_{24h} , Δ_{48h} , Δ_{72h} , \min_{24h} , $\Delta_{\text{mean}_{24h}}$	Hour, day of the week, month	Temperature, occupancy, degree day, daily meals, rooms
4	Reading, Δ_{1h}^b , Δ_{2h}^b , Δ_{3h}^b , Δ_{24h}^b , Δ_{48h}^b , Δ_{72h}^b , \min_{24h} , $\Delta_{\text{mean}_{24h}^b}$	Hour, day of the week, month	Temperature, occupancy, degree day, daily meals, rooms

6.3. Results of the Tests Performed with the Isolation Forest

The Sklearn library (version 1.1.1) [63] was used to perform the computational tests of the IF algorithm. In terms of algorithm settings, a grid search was performed on the hyperparameters $\text{contamination} \in \{0.008, 0.01, 0.03, 0.05, 0.1\}$, $\text{bootstrap} \in \{\text{True}, \text{False}\}$, and $\text{max_samples} \in \{\text{'auto'}, 0.05, 0.1, 0.3, 0.5\}$. For the remaining hyperparameters (see the IF class documentation <https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.IsolationForest.html>, accessed on 27 May 2022), the default values of the library were considered. The best results obtained by the IF algorithm are summarized in Table 5. From the results, it can be concluded that, according to the metrics defined before, the algorithm performed well in the detection of unknown points ($S_1 = 0.99$) for all meters, that is, the number of unidentified readings classified as anomalous remained within the tolerance range. The same happened also in early-stage anomaly detection (S_2). An exception was the anomaly “ S_2 Anomaly 2” in the “Electricity 1” meter, which could not be identified, penalizing its S_{final} , reaching only 0.73 points on the scale.

Analyzing the results, it appears that the model, in a real context, needs to have the parameters and input variables adjusted according to each meter. In terms of variables, it was found that three of the best results are associated with scenarios where hotel and environmental variables (as expressed in Table 5) were present. This indicates their relevance and the fact that they should be considered whenever possible. When choosing the set of variables for the training base of the models, it was clear that the readings from meters, temporal variables, and \min_{24h} seem to improve the performance of the algorithm. The results were unclear regarding the integration of the variables Δ_X and $\Delta_{\text{average}_{24h}}$, particularly regarding whether they should be considered with the calculated difference or in the binary perspective of consumption increase/decrease. Therefore, in order to make a decision, a critical analysis of the situation was carried out and it was understood that a binary approach could bring some loss of sensitivity, specifically placing large and small variations in the same proportion. Regarding the hyperparameters, it is not possible to draw a clear conclusion, given the diversity of data types considered, so keeping pre-established parameters seems to be a viable solution. In the context of use, the parameter contamination must be adjusted according to the results by the system manager. It should also be noted that several of the simulations which were carried out took an average of about 30 s to complete, which is relatively fast considering the number of samples used. This proves that the IF algorithm requires little computational cost, confirming the results obtained for examples by de Santis and Costa [47].

Table 5. Best results for each meter with the IF algorithm, according to the proposed metric, among all scenarios and considering the mentioned grid-search.

Meters	Scenario	contamination	max_samples	bootstrap	S_1	S_2 Anomaly 1	S_2 Anomaly 2	S_2 Anomaly 3	S_2 Anomaly 4	S_{final}
Water	4	0.008	'auto'	True	0.99	0.99	0.78	0.99	0.99	0.94
Electricity 1	2	0.05	0.3	False	0.98	0.99	0.00	0.97	0.96	0.73
Electricity 2	3	0.1	'auto'	False	0.94	0.99	0.99	0.99	0.99	0.94
Gas	3	0.05	0.1	True	0.96	0.99	0.90	0.98	0.99	0.93

6.4. Results of the Tests Performed with the Variational Autoencoder

The Keras computational library (version 2.7.0) [64] was used to implement the VAE architecture, as explained in Section 3.1.2. Input data were grouped into sequences with a dimension $t = 24$, and anomaly detection was performed based on the reconstruction error, using a sliding window [65,66]. Given the computational restrictions in the tests of this model, only the width of the sliding window was changed (24, 48, and 72), because it soon was understood that simulations could take approximately 8 h for 1500 training cycles.

The model performed well on three types of meters, as shown in Table 6. The exception was the “Water” meter, with a poor performance in both the detection of unknown points ($S_1 = 0.79$) and in the detection of anomalies, with one of them not being detected and the other being detected late. This led to a final score of 0.57. There were also problems with the model convergence during the tests in some scenarios and features, which we were unable to understand.

The scenarios with the best performance in this model were 1 and 2, which in both cases did not use hotel variables. The results were not conclusive with regard to the width of the sliding window to be used. Although two of the best results used a sliding window of 24, one of these meters was “Water”, which did not have a good performance and the remaining meters used a width of 48 and 72. For that, this parameter should eventually be user-adaptable.

Table 6. Best results for each meter with the VAE algorithm, according to the proposed metric, among all scenarios and considering the mentioned grid-search.

Meters	Scenario	Sliding window	S_1	S_2 Anomaly 1	S_2 Anomaly 2	S_2 Anomaly 3	S_2 Anomaly 4	S_{final}
Water	1	24	0.79	0.99	0.00	0.99	0.90	0.57
Electricity 1	1	24	0.85	0.99	0.99	0.99	0.99	0.85
Electricity 2	2	48	0.96	0.99	0.99	0.99	0.99	0.96
Gas	2	72	0.99	0.99	0.99	0.85	0.99	0.95

6.5. Comparison between the Algorithms

Comparing the VAE results with the ones of IF, it was concluded that the latter would be a better solution for the problem studied, because, according to the obtained results, it presented more stable outcomes in terms of the final score. For example, the average value of S_{final} for the IF algorithm was 0.89, while it was 0.83 for the VAE. It was found, however, that the solution based on the VAE algorithm, when all the proposed anomalies

were detected, was able to surpass the performance of the IF, for the “Electricity 2” and “Gas” meters. However, in the “Water” meter, the VAE algorithm had the worst score of the presented results. There were also problems of convergence in some scenarios, as mentioned. This indicated the possibility of the algorithm not being sufficiently robust when applied to solutions such as the ones analyzed herein, which needs to be as adaptable as possible over an uncontrolled environment. It was also verified during the tests that the training of VAE models is time-consuming and computationally heavy. This individual computational cost, associated with a high number of hotels, meters, and the more or less frequent (re-)training of the algorithms, could raise additional problems regarding the hardware requirements of the global system. Another reason that did not support the VAE was the large amount of history that must exist for these models to be functional and robust, which could be an impediment to making proper anomaly detection for new meters.

7. System Proposal

Concluding the tests with the algorithms, the structure of the intelligent system to identify anomalies in hotel consumption is depicted in Figure 16. It is composed of two main modules: (i) the data correction module, explained in Section 5, which handles data problems and guarantees that the data used in the next phase have quality and liability; and (ii) the alarmistic model, with the IF algorithm as the backbone of the model, to analyze and identify possible consumption anomalies in the hotel.

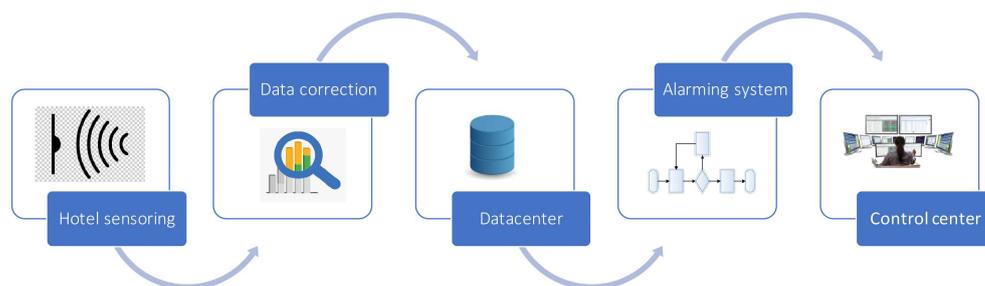


Figure 16. Proposal of architecture for the intelligent system to detect consumption anomalies in a hotel.

In more detail, the hotel sensing provides a stream of data that passes through the data correction module before being stored in a data center. Then, the alarming system looks for new entries and runs the anomaly detection model to find anomalous readings. If found, those anomalous readings trigger an alarm in the control center. We should notice that changes made by the correction module also trigger alarms in the control center, as some actions can be activated from them (e.g., correct malfunctioning installations).

To illustrate the results obtained by the proposed system working under real conditions, in Figure 17, two detections of anomalous scenarios are represented, which are associated with two meters of a hotel. These situations were reported by the promoter and validated by the system. In the first scenario (top image), the detected drop in consumption was associated with a machinery failure, and after some days, another abnormal event (not specified) was also detected. In the bottom image, an increase in consumption was also detected, associated with a water pump of the hotel, which started working at a new unnecessary regime.

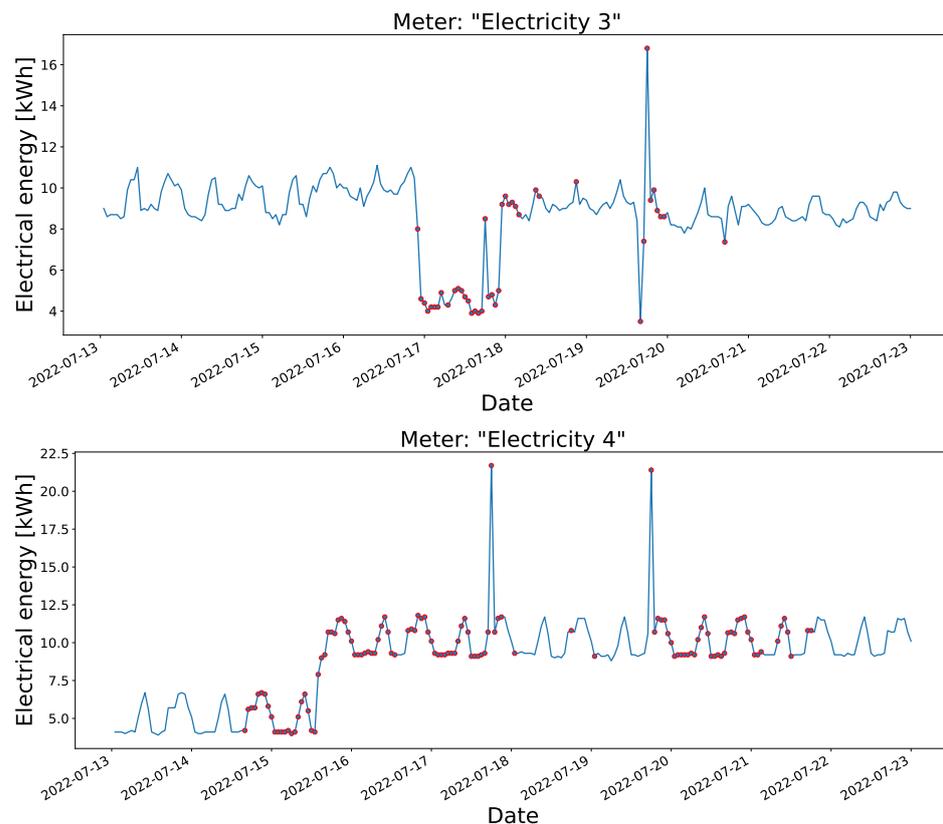


Figure 17. Real anomaly detection scenarios detected by the proposed system. The first scenario (**top**) is associated with a machinery failure, and the second one (**bottom**) is related to one water pump that started working on a new unnecessary regime.

8. Conclusions and Future Work

This case study aimed to compare IF and VAE algorithms and to choose the best approach to develop and implement an anomaly detection system for hotel consumption. This system is intended to avoid unnecessary consumption or even detect malfunctions in the installations.

During the study carried out on the data, there were some problems in the provided readings, which could interfere with the performance of the platform, such as: (i) problematic and (ii) missing readings. The solution found was the data correction module with statistical approaches, like the IQR method and LWMA.

To evaluate the performance of the detection of anomalies in an unsupervised time series by IF and VAE algorithms, a performance metric was defined. Comparing the results obtained between the two algorithms, it was concluded that generally, the IF algorithm gave a better answer to the problem in the study, as it obtained a better balance in the test results compared to the VAE. For example, the average of the S_{final} of the IF algorithm was 0.89, while that of the VAE was 0.83. VAE, on the other hand, proved to be able to provide good detection results, including better results than IF in two situations, which may indicate the potential to solve problems, where generalization, the easy adaptability of the algorithm, and limited computational resources are not considered. In conclusion, the IF algorithm belongs to the ML algorithms of the unsupervised type, which means that it performs an analysis of unclassified data. This algorithm was able to answer the specific needs of this problem, demonstrating its capacity to detect artificially introduced anomalies, to have computationally less undemanding training, and to be very adaptable to the different types of variables that are available for analysis.

In the future, it will be interesting to further explore the data processing component, evaluating new approaches and improving the actual solution to remove the problematic

readings. Furthermore, testing alternative anomaly detection approaches could provide more accurate methods. For example, it could be interesting to employ active learning to overcome the unlabeled problem of the datasets, or use attention mechanisms in the deep learning approach to focus on the relevant parts of the time series sequences. Furthermore, the metric proposed to evaluate anomaly detection algorithms using unlabeled time series datasets has space to be improved.

Author Contributions: Conceptualization, T.M., P.J.S.C., J.M. and J.R.; methodology, T.M., P.J.S.C., J.M. and J.R.; software, T.M.; validation, J.R.; formal analysis, T.M., P.J.S.C. and J.M.; investigation, T.M., P.J.S.C. and J.M.; resources, J.R.; data curation, T.M. and J.R.; writing—original draft preparation, T.M., P.J.S.C. and J.M.; writing—review and editing, T.M., P.J.S.C. and J.M.; visualization, T.M., P.J.S.C., J.M. and J.R.; supervision, P.J.S.C., J.M. and J.R.; project administration, J.M. and J.R.; funding acquisition, J.M. and J.R. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the European Union through CRESC ALGARVE 2020, Portugal 2020, and by the Portuguese Foundation for Science and Technology, project UIDB/50009/2020—LARSyS.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: We would like to thank the company “Âmago—Energia Inteligente” for its involvement in the conceptualization, for providing the data which made this study possible, and for all the support during its realization.

Conflicts of Interest: The authors declare no conflict of interest. The funders played no role in the design of the study; in the collection, analyses, or interpretation of the data; in the writing of the manuscript; or in the decision to publish the results.

Abbreviations

The following abbreviations are used in this manuscript:

AE	Autoencoder
CTMS	Centralized Technical Management System
DL	Deep Learning
EDA	Exploratory Data Analysis
IF	Isolation Forest
IQR	Interquartile Range
LSTM	Long Short-Term Memory
LWMA	Linear Weighted Moving Average
ML	Machine Learning
NAB	Numenta Anomaly Benchmark
SDM	Standard Deviation Method
VAE	Variational Autoencoder

References

1. International Energy Agency. Review 2021—Assessing the Effects of Economic Recoveries on Global Energy Demand and CO₂ Emissions in 2021 Global Energy. 2021. Available online: <https://www.iea.org/reports/global-energy-review-2021> (accessed on 15 December 2021).
2. United Nations. Theme Report on Energy Transition towards the Achievement of SDG 7 and Net-Zero Emissions, 2021. Available online: https://www.un.org/sites/un2.un.org/files/2021-twg_2-062321.pdf (accessed on 23 March 2022).
3. Associação Portuguesa de Energias Renováveis. APREN—Boletim Dezembro 2021. 2022. Available online: <https://www.apren.pt/contents/publicationsreportcarditens/boletim-renovaveis-dezembro-2021.pdf> (accessed on 16 January 2022). (In Portuguese)
4. International Energy Agency. Electric Cars Fend off Supply Challenges to More Than Double Global Sales. 2022. Available online: <https://www.iea.org/commentaries/electric-cars-fend-off-supply-challenges-to-more-than-double-global-sales> (accessed on 23 March 2022).
5. Holmberg, K.; Erdemir, A. The impact of tribology on energy use and CO₂ emission globally and in combustion engine and electric cars. *Tribol. Int.* **2019**, *135*, 389–396. [CrossRef]

6. Klöckner, C.A.; Nayum, A.; Mehmetoglu, M. Positive and negative spillover effects from electric car purchase to car use. *Transp. Res. Part D Transp. Environ.* **2013**, *21*, 32–38. [[CrossRef](#)]
7. Gul, M.S.; Patidar, S. Understanding the energy consumption and occupancy of a multi-purpose academic building. *Energy Build.* **2015**, *87*, 155–165. [[CrossRef](#)]
8. Himeur, Y.; Alsalemi, A.; Bensaali, F.; Amira, A. Smart power consumption abnormality detection in buildings using micromoments and improved K-nearest neighbors. *Int. J. Intell. Syst.* **2021**, *36*, 2865–2894. [[CrossRef](#)]
9. Fenza, G.; Gallo, M.; Loia, V. Drift-Aware Methodology for Anomaly Detection in Smart Grid. *IEEE Access* **2019**, *7*, 9645–9657. [[CrossRef](#)]
10. Araya, D.B.; Grolinger, K.; ElYamany, H.F.; Capretz, M.A.; Bitsuamlak, G. An ensemble learning framework for anomaly detection in building energy consumption. *Energy Build.* **2017**, *144*, 191–206. [[CrossRef](#)]
11. Eras, J.J.C.; Santos, V.S.; Gutiérrez, A.S.; Álvarez Guerra Plasencia, M.; Haeseldonckx, D.; Vandecasteele, C. Tools to improve forecasting and control of the electricity consumption in hotels. *J. Clean. Prod.* **2016**, *137*, 803–812. [[CrossRef](#)]
12. Deng, S.M.; Burnett, J. A study of energy performance of hotel buildings in Hong Kong. *Energy Build.* **2000**, *31*, 7–12. [[CrossRef](#)]
13. Cook, A.A.; Mısırlı, G.; Fan, Z. Anomaly detection for IoT time-series data: A survey. *IEEE Internet Things J.* **2019**, *7*, 6481–6494. [[CrossRef](#)]
14. Dunning, T.; Friedman, E. *Practical Machine Learning: A New Look at Anomaly Detection*; O'Reilly Media, Inc.: Sebastopol, CA, USA, 2014.
15. Chandola, V.; Banerjee, A.; Kumar, V. Anomaly detection: A survey. *ACM Comput. Surv. (CSUR)* **2009**, *41*, 1–58. [[CrossRef](#)]
16. Zheng, L.; Liu, G.; Yan, C.; Jiang, C.; Zhou, M.; Li, M. Improved TrAdaBoost and its application to transaction fraud detection. *IEEE Trans. Comput. Soc. Syst.* **2020**, *7*, 1304–1316. [[CrossRef](#)]
17. Abdallah, A.; Maarof, M.A.; Zainal, A. Fraud detection system: A survey. *J. Netw. Comput. Appl.* **2016**, *68*, 90–113. [[CrossRef](#)]
18. Shen, A.; Tong, R.; Deng, Y. Application of classification models on credit card fraud detection. In Proceedings of the 2007 International Conference on Service Systems and Service Management, Chengdu, China, 9–11 June 2007; IEEE: New York, NY, USA, 2007; pp. 1–4.
19. Deecke, L.; Vandermeulen, R.; Ruff, L.; Mandt, S.; Kloft, M. Image anomaly detection with generative adversarial networks. In Proceedings of the Joint European Conference on Machine Learning and Knowledge Discovery in Databases, Dublin, Ireland, 10–14 September 2018; Springer: Berlin/Heidelberg, Germany, 2018; pp. 3–17.
20. Bao, Y.; Tang, Z.; Li, H.; Zhang, Y. Computer vision and deep learning-based data anomaly detection method for structural health monitoring. *Struct. Health Monit.* **2019**, *18*, 401–421. [[CrossRef](#)]
21. Tang, Z.; Chen, Z.; Bao, Y.; Li, H. Convolutional neural network-based data anomaly detection method using multiple information for structural health monitoring. *Struct. Control Health Monit.* **2019**, *26*, e2296. [[CrossRef](#)]
22. Zhou, P.; Zhou, G.; Wang, H.; Wang, D.; He, Z. Automatic detection of industrial wire rope surface damage using deep learning-based visual perception technology. *IEEE Trans. Instrum. Meas.* **2020**, *70*, 1–11. [[CrossRef](#)]
23. Li, D.; Ho, S.C.M.; Song, G.; Ren, L.; Li, H. A review of damage detection methods for wind turbine blades. *Smart Mater. Struct.* **2015**, *24*, 033001. [[CrossRef](#)]
24. Himeur, Y.; Ghanem, K.; Alsalemi, A.; Bensaali, F.; Amira, A. Artificial intelligence based anomaly detection of energy consumption in buildings: A review, current trends and new perspectives. *Appl. Energy* **2021**, *287*, 116601. [[CrossRef](#)]
25. Omar, S.; Ngadi, A.; Jebur, H.H. Machine Learning Techniques for Anomaly Detection: An Overview. *Int. J. Comput. Appl.* **2013**, *79*, 33–41. [[CrossRef](#)]
26. Cervantes, J.; Garcia-Lamont, F.; Rodríguez-Mazahua, L.; Lopez, A. A comprehensive survey on support vector machine classification: Applications, challenges and trends. *Neurocomputing* **2020**, *408*, 189–215. [[CrossRef](#)]
27. Yan, W. Detecting gas turbine combustor anomalies using semi-supervised anomaly detection with deep representation learning. *Cogn. Comput.* **2020**, *12*, 398–411. [[CrossRef](#)]
28. Akcay, S.; Atapour-Abarghouei, A.; Breckon, T.P. Ganomaly: Semi-supervised anomaly detection via adversarial training. In Proceedings of the Asian Conference on Computer Vision, Perth, Australia, 2–6 December 2018; Springer: Berlin/Heidelberg, Germany, 2018; pp. 622–637.
29. Zong, B.; Song, Q.; Min, M.R.; Cheng, W.; Lumezanu, C.; Cho, D.; Chen, H. Deep autoencoding gaussian mixture model for unsupervised anomaly detection. In Proceedings of the International Conference on Learning Representations, Vancouver, BC, Canada, 30 April–3 May 2018.
30. Munir, M.; Siddiqui, S.A.; Dengel, A.; Ahmed, S. DeepAnT: A deep learning approach for unsupervised anomaly detection in time series. *IEEE Access* **2018**, *7*, 1991–2005. [[CrossRef](#)]
31. Xiong, Z.; Zhu, D.; Liu, D.; He, S.; Zhao, L. Anomaly Detection of Metallurgical Energy Data Based on iForest-AE. *Appl. Sci.* **2022**, *12*, 9977. [[CrossRef](#)]
32. Kaymakci, C.; Wenninger, S.; Sauer, A. Energy Anomaly Detection in Industrial Applications with Long Short-term Memory-based Autoencoders. *Procedia CIRP* **2021**, *104*, 182–187. [[CrossRef](#)]

33. Himeur, Y.; Alsalemi, A.; Bensaali, F.; Amira, A. Detection of Appliance-Level Abnormal Energy Consumption in Buildings Using Autoencoders and Micro-moments. In Proceedings of the 5th International Conference on Big Data and Internet of Things, Beijing, China, 16–18 December 2022; Springer International Publishing: Berlin/Heidelberg, Germany, 2022; pp. 179–193. [[CrossRef](#)]
34. Apostol, E.S.; Truică, C.O.; Pop, F.; Esposito, C. Change Point Enhanced Anomaly Detection for IoT Time Series Data. *Water* **2021**, *13*, 1633. [[CrossRef](#)]
35. Liu, X.; Ding, Y.; Tang, H.; Xiao, F. A data mining-based framework for the identification of daily electricity usage patterns and anomaly detection in building electricity consumption data. *Energy Build.* **2021**, *231*, 110601. [[CrossRef](#)]
36. Lipčák, P.; Macak, M.; Rossi, B. Big Data Platform for Smart Grids Power Consumption Anomaly Detection. In Proceedings of the 2019 Federated Conference on Computer Science and Information Systems (FedCSIS), Leipzig, Germany, 1–4 September 2019; pp. 771–780. [[CrossRef](#)]
37. Chiosa, R.; Piscitelli, M.S.; Capozzoli, A. A Data Analytics-Based Energy Information System (EIS) Tool to Perform Meter-Level Anomaly Detection and Diagnosis in Buildings. *Energies* **2021**, *14*, 237. [[CrossRef](#)]
38. Al-amri, R.; Murugesan, R.K.; Man, M.; Abdulateef, A.F.; Al-Sharafi, M.A.; Alkahtani, A.A. A Review of Machine Learning and Deep Learning Techniques for Anomaly Detection in IoT Data. *Appl. Sci.* **2021**, *11*, 5320. [[CrossRef](#)]
39. Pereira, J.; Silveira, M. Unsupervised anomaly detection in energy time series data using variational recurrent autoencoders with attention. In Proceedings of the 2018 17th IEEE International Conference on Machine Learning and Applications (ICMLA), Orlando, FL, USA, 17–20 December 2018; IEEE: New York, NY, USA, 2018; pp. 1275–1282.
40. Zhou, C.; Paffenroth, R.C. Anomaly detection with robust deep autoencoders. In Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax, NS, Canada, 13–17 August 2017; Part F129685; pp. 665–674. [[CrossRef](#)]
41. Chen, Z.; Yeo, C.K.; Lee, B.S.; Lau, C.T. Autoencoder-based network anomaly detection. In Proceedings of the Wireless Telecommunications Symposium, Phoenix, AZ, USA, 17–20 April 2018; pp. 1–5. [[CrossRef](#)]
42. Yao, R.; Liu, C.; Zhang, L.; Peng, P. Unsupervised anomaly detection using variational auto-encoder based feature extraction. In Proceedings of the 2019 IEEE International Conference on Prognostics and Health Management (ICPHM), San Francisco, CA, USA, 17–20 June 2019; IEEE: New York, NY, USA, 2019; pp. 1–7.
43. Liu, F.T.; Ting, K.M.; Zhou, Z.H. Isolation forest. In Proceedings of the 2008 Eighth IEEE International Conference on Data Mining, Pisa, Italy, 15–19 December 2008; IEEE: New York, NY, USA, 2008; pp. 413–422.
44. AbuAlghanam, O.; Alazzam, H.; Alhenawi, E.; Qatawneh, M.; Adwan, O. Fusion-based anomaly detection system using modified isolation forest for internet of things. *J. Ambient. Intell. Humaniz. Comput.* **2022**, 1–15. [[CrossRef](#)]
45. Ding, Z.; Fei, M. An Anomaly Detection Approach Based on Isolation Forest Algorithm for Streaming Data using Sliding Window. *IFAC Proc. Vol.* **2013**, *46*, 12–17. [[CrossRef](#)]
46. Puggini, L.; McLoone, S. An enhanced variable selection and Isolation Forest based methodology for anomaly detection with OES data. *Eng. Appl. Artif. Intell.* **2018**, *67*, 126–135. [[CrossRef](#)]
47. de Santis, R.B.; Costa, M.A. Extended isolation forests for fault detection in small hydroelectric plants. *Sustainability* **2020**, *12*, 6421. [[CrossRef](#)]
48. Bank, D.; Koenigstein, N.; Giryas, R. Autoencoders. *arXiv* **2020**, arXiv:2003.05991
49. Baldi, P. Autoencoders, unsupervised learning, and deep architectures. In Proceedings of ICML Workshop on Unsupervised and Transfer Learning, JMLR Workshop and Conference Proceedings, Bellevue, Washington, DC, USA, 27 June 2012; pp. 37–49.
50. Yu, Y.; Si, X.; Hu, C.; Zhang, J. A review of recurrent neural networks: LSTM cells and network architectures. *Neural Comput.* **2019**, *31*, 1235–1270. [[CrossRef](#)]
51. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780. [[CrossRef](#)]
52. Uchida, K.; Shirakawa, S.; Akimoto, Y. Analysis of information geometric optimization with isotropic Gaussian distribution under finite samples. In Proceedings of Genetic and Evolutionary Computation Conference, Kyoto, Japan, 15–19 July 2018; pp. 897–904.
53. An, J.; Cho, S. Variational autoencoder based anomaly detection using reconstruction probability. *Spec. Lect. IE* **2015**, *2*, 1–18.
54. Xu, H.; Chen, W.; Zhao, N.; Li, Z.; Bu, J.; Li, Z.; Liu, Y.; Zhao, Y.; Pei, D.; Feng, Y.; et al. Unsupervised anomaly detection via variational auto-encoder for seasonal kpis in web applications. In Proceedings of the 2018 World Wide Web Conference, Lyon, France, 23–27 April 2018; pp. 187–196.
55. Lavin, A.; Ahmad, S. Evaluating real-time anomaly detection algorithms—the Numenta anomaly benchmark. In Proceedings of the 2015 IEEE 14th International Conference on Machine Learning and Applications (ICMLA), Miami, FL, USA, 9–11 December 2015; IEEE: New York, NY, USA, 2015; pp. 38–44.
56. Brownlee, J. *Data Preparation for Machine Learning: Data Cleaning, Feature Selection, and Data Transforms in Python*; Machine Learning Mastery: Vermont, VIC, Australia, 2020.
57. Tukey, J.W. *Exploratory Data Analysis*; Addison Wesley-Publishing Company: Boston, MA, USA, 1977.
58. Moustafa, N.; Creech, G.; Sitnikova, E.; Keshk, M. Collaborative anomaly detection framework for handling big data of cloud computing. In Proceedings of the 2017 Military Communications and Information Systems Conference (MilCIS), Canberra, Australia, 14–16 November 2017; IEEE: New York, NY, USA, 2017; pp. 1–6.

59. Saradjian, M.; Akhoondzadeh, M. Thermal anomalies detection before strong earthquakes ($M > 6.0$) using interquartile, wavelet and Kalman filter methods. *Nat. Hazards Earth Syst. Sci.* **2011**, *11*, 1099–1108. [[CrossRef](#)]
60. Patil, N.; Das, D.; Pecht, M. Anomaly detection for IGBTs using Mahalanobis distance. *Microelectron. Reliab.* **2015**, *55*, 1054–1059. [[CrossRef](#)]
61. Hassani, H.; Kalantari, M.; Ghodsi, Z. Evaluating the Performance of Multiple Imputation Methods for Handling Missing Values in Time Series Data: A Study Focused on East Africa, Soil-Carbonate-Stable Isotope Data. *Stats* **2019**, *2*, 457–467. [[CrossRef](#)]
62. Demirhan, H.; Renwick, Z. Missing value imputation for short to mid-term horizontal solar irradiance data. *Appl. Energy* **2018**, *225*, 998–1012. [[CrossRef](#)]
63. Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.
64. Chollet, F.; et al. Keras. 2015. Available online: <https://github.com/fchollet/keras> (accessed on 15 April 2022)
65. Aminikhanghahi, S.; Cook, D.J. A survey of methods for time series change point detection. *Knowl. Inf. Syst.* **2017**, *51*, 339–367. [[CrossRef](#)]
66. Hota, H.; Handa, R.; Shrivastava, A. Time series data prediction using sliding window based RBF neural network. *Int. J. Comput. Intell. Res.* **2017**, *13*, 1145–1156.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.