


RESEARCH

Open Access



Is bigger always better? A controversial journey to the center of machine learning design, with uses and misuses of big data for predicting water meter failures

Marco Rocchetti^{1*} , Giovanni Delnevo¹, Luca Casini¹ and Giuseppe Cappiello²

*Correspondence:

marco.rocchetti@unibo.it

¹ Department of Computer

Science and Engineering,

University of Bologna, Via

Mura Anteo Zamboni 7,

40127 Bologna, Italy

Full list of author information

is available at the end of the

article

Abstract

In this paper, we describe the design of a machine learning-based classifier, tailored to predict whether a water meter will fail or need a replacement. Our initial attempt to train a recurrent deep neural network (RNN), based on the use of 15 million of readings gathered from 1 million of mechanical water meters, spread throughout Northern Italy, led to non-positive results. We learned this was due to a lack of specific attention devoted to the quality of the analyzed data. We, hence, developed a novel methodology, based on a new semantics which we enforced on the training data. This allowed us to extract only those samples which are representative of the complex phenomenon of defective water meters. Adopting such a methodology, the accuracy of our RNN exceeded the 80% threshold. We simultaneously realized that the new training dataset differed significantly, in statistical terms, from the initial dataset, leading to an apparent paradox. Thus, with our contribution, we have demonstrated how to reconcile such a paradox, showing that our classifier can help detecting defective meters, while simplifying replacement procedures.

Keywords: Smart data, Machine learning design, Human-machine-bigdata interaction loop, Human-in-the-loop methods, Water metering and consumption

Introduction

If modern artificial intelligence (AI) comes often misunderstood, this is mainly due to the fact that, historically, it is solely tied to the way human brains work and think. New machine learning (ML) algorithms, instead, learn now by processing massive piles of data. This process enables machines to adapt to real-world situations, as well as to propose suggestions on how to classify and interpret a variety of different real phenomena. Simply speaking, the deployment of modern ML systems into critical applications is directly influenced by the way training data are organized and modeled [1–3]. Hence, while those modern algorithms rapidly sift through huge datasets, loaded with millions of information, a thoughtfully designed AI, beyond its ML-based core, should never disregard the fact that algorithms that learn are, for now, just another form of machine instruction, still guided and influenced by the potential and the limitations that training data carry with them. In other words, even when we train algorithms to learn basic associations that can then be used to approximate,

or infer, some aspects of a given process, crucial remains the process of harnessing those piles of data into realistic findings. No matter how much sophisticated is the algorithm that will analyze a dataset, equally critical is the statistical validity, the sense, the references, the subtle implications, in one simple word: the semantics, being inherent in those data [4–6].

This exactly was the case of our controversial experience with a huge real-world dataset, fed with over 15 million water meter readings, supplied by a company that distributes water over a large area in Northern Italy. In this context, we were asked to design an ML-based intelligent classifier, able to predict if a water meter fails/needs disassembly, based on a history of water consumption measurements, thus minimizing the number of technical interventions performed by human operators for maintenance and repair.

Indeed, our initial attempts to train a *recurrent neural network*, without a specific attention to the quality, and to the limitations, of those data used for training, led to unexpected and negative prediction outcomes. Along this line of applied research, this paper reports on the combination of actions we had to take, in terms of statistical tests and data semantics to be enforced, to extrapolate from that large initial database just those training data that could make a sense, as well as that could safely represent the complex statistical phenomenon under observation, with the final target of training a machine able to predict a failure of a water meter, not only in a dataset but also in a real practical case.

As a result of these data modeling and re-organization activities, and upon completion of the training process on a safe subset of the initial dataset, our classifier upheld its performance level, from approx. 60% to about 80–90%, in terms of prediction accuracy. Nonetheless, this performance outcome came with the paradox of a statistical transformation of the initial dataset, thus confirming one of our research conjecture in this field: the need for millions of training data can become a non-issue, as compared to a paltrier training set that makes, instead, a learning algorithm much more realistically applicable.

Paper organization

The remainder of this paper is structured as follows. In the section devoted to “[Related work](#)”, we present the research background on which our study relies on. In “[Methodology](#)” section, we discuss the methods through which the initial dataset was remodeled, to make it adequate to be used for training several learning algorithms. In the section devoted to describing “[Results](#)”, instead, we first illustrate the accuracy of the results we have obtained after training an intelligent classifier able to predict water meter failures, and then we compare them with the performances that can be achieved with alternative algorithms. At the end of “[Results](#)” section, we put a focus on a statistical paradox it has emerged after our data transformation activities. The section devoted to “[Discussion](#)” supplies a reconciliation of that paradox, along with a practical guide on how to put to good use our classifier. The final section provides “[Conclusions](#)” of the paper.

Related work

This section is split over two different parts. The first one discusses other studies that have already done in the specific domain of automatic methods for detecting faulty water meters. The second one, instead, illustrates the negative effects we can incur due to a lack of attention in data preparation while instructing machine learning algorithms.

Detecting water meter failures

While we are plenty of papers in the literature that employ complex statistical methods, or machine learning algorithms, for individuating anomalies like a leakage or a failure, in water distribution pipelines [7, 8], there is a not surprising scarcity of papers that discuss methods for detecting anomalies in water meters. *Pour cause*: so far, in fact, smart metering has come into the scene for utilities different from water, like energy and gas, since these latter resources are considered more expensive, in general.

This motivates the fact why there are a lot of mechanical water meters around, whose main characteristic, different from electrical meters, is that of providing *fewer* and *less frequent* readings over time. Hence, even if the number of mechanical meters installed is still high, representing a cheap and well-tested solution, they pose a problem to all the initiatives that are based on machine learning [9]. Indeed, their readings are rare (2/3/4 times per year) and are to be read by a human operator, thus resulting in many imprecisions. Due to this fact, some of the most relevant papers that illustrate methods that face the problem of detecting faulty water meters, on the basis of an analysis of the amount of consumed water, still resort to traditional approaches, disregarding machine learning.

For example, Roberts and Monk developed a simple algorithm that individuates possible anomalies [10], occurring at a given water meter, when a decreasing trend in water consumption is observed along a series of readings which is updated just quarterly.

Monedero et al. [11], instead, propose an approach to detect tampering activities in mechanical water meters that employs a very basic statistical analysis for identifying:

- either a low rate in water consumption,
- or a sudden stoppage of that consumption,
- or simply a decreasing consumption trend.

What is relevant, here again, is the fact that the use of data (readings), that can be, large in quantity yet rare in frequency, does not allow for the use of modern machine learning-based methods.

For sure, the advent of electrical water meters, along with telemetry that can provide water consumption readings on a per hour basis, could significantly alter this picture.

In this unfortunate scenario, our challenge has been precisely that of trying to use learning algorithms, even if trained with data coming from readings of traditional mechanical meters, believing that the large amount of data available, spanning over multiple years and involving more than 1 million meters, could balance the negative effects of the low frequency in reading.

Inadequacy of the datasets

In this subsection, instead, we are concerned with the problem that, while machine learning techniques analyze datasets that are very large and expensive, it is often the case when results come up that are inaccurate, or even wrong. Oversimplifying a complex scenario, there are an exaggerated anxiety and worry of developing specific learning algorithms that search through a vast amount of data until they recognize a pattern that finally exists only in (a portion of) that dataset, and not in the reality [12].

As a consequence of these considerations, if we want to discover results that stand the test of time, adequate attention is to be devoted to all the data preparation, cleaning and transformation activities that must follow their initial acquisition. Disregarding, or simply underestimating, these factors means crystallize data inconsistencies and impurities into a shapeless structure that will be inadequate for supporting correct evidence-based decisions.

Drawing upon scientific literature, among all the possible cases we could cite in support of our ideas, we report here just three different examples, where a lack of attention on data used to instruct intelligent machines resulted into negative consequences, as well as into effects diametrically opposed to what we would expect.

The first example is the paradigmatic case discussed by Buolamwini and Gebru in [13]. After a careful assessment, a gender classification system, based on a facial analysis dataset, came up not to be balanced with respect to gender and skin type. It was found out in fact that the most misclassified group was that of the darker-skinned females, with misclassification rates ranging from 20.8 to 34.7%, while, instead, the error rate for the lighter-skinned males group stabilized on around 0.8%. Such result was the direct consequence of the dataset that was used for training that system. An *ex-post* accurate analysis of the dataset simply revealed that it was biased, as overwhelmingly comprised of lighter-skinned subjects.

Another similar example is reported by Bolukbasi and coauthors in [14] that treats word representations. Specifically, with the term *word embedding* is intended a representation of words under the form of vectors, commonly used for natural language processing. The popularity of this kind of word representation comes from its ability to capture semantic relationships among words, measurable as linear distances between vectors [15]. An experiment was conducted that tried to train a ML-based implementation of a word embedding representation, only using articles taken from the Google News service. To simplify this complex matter, we only say that the main result of this specific implementation of a word embedding representation was to let emerge a dramatic case of sex stereotype. For example, the role played by a surgeon was always associated with a masculine subject, while at the opposite the functions of a nurse were always perceived as carried out by a feminine subject. The motivation for the emergence of this gender stereotype was rooted in that specific dataset, as an expensive assessment activity demonstrated. Further, an additional specific procedure was developed that de-biased that representation to the point it finally became gender-neutral.

A final example is drawn from a medical context where a case is reported discussing on a machine trained to learn a prognostic model used to predict adequate medical treatments for patients affected by pneumonia [16]. Surprisingly, upon completion of the training phase, the machine had learnt that patients suffering from both pneumonia and asthma were to be considered at a lower risk of death, if compared with those who were afflicted by just pneumonia. The motivation why asthma was (erroneously) considered by the machine (almost) as a *protective factor* against the negative effects of pneumonia was pretty clear after an *ex-post* analysis of the dataset on which the machine was instructed. The reason goes as follows. Patients diagnosed with pneumonia, and with a history of asthma, are typically admitted to more intensive care, leading on average to more rapid healing, with respect to patients diagnosed with just pneumonia.

Unfortunately, the training dataset was constructed disregarding that relevant fact, with the final paradox of a machine that misinterpreted asthma as a *protective* variable in that specific domain.

After all these preliminary examples and consequent discussions, it is now the time to start our controversial journey with a real case study in the field of water distribution and relative measurement procedures. We will travel through machine learning techniques, statistical procedures and data re-organization activities, towards achieving final results we consider important especially because how uncertain, they can be, is measured with precision and discussed at length.

Methodology

An important water supply company that distributes water over a large area in Northern Italy asked us to deploy an AI model in the field of water metering; the final target being that of training a machine able to predict when a water meter fails/needs disassembly, based on a set of meter readings taken over the past. An important additional requirement was that of minimizing the number of consecutive readings to be used for such prediction.

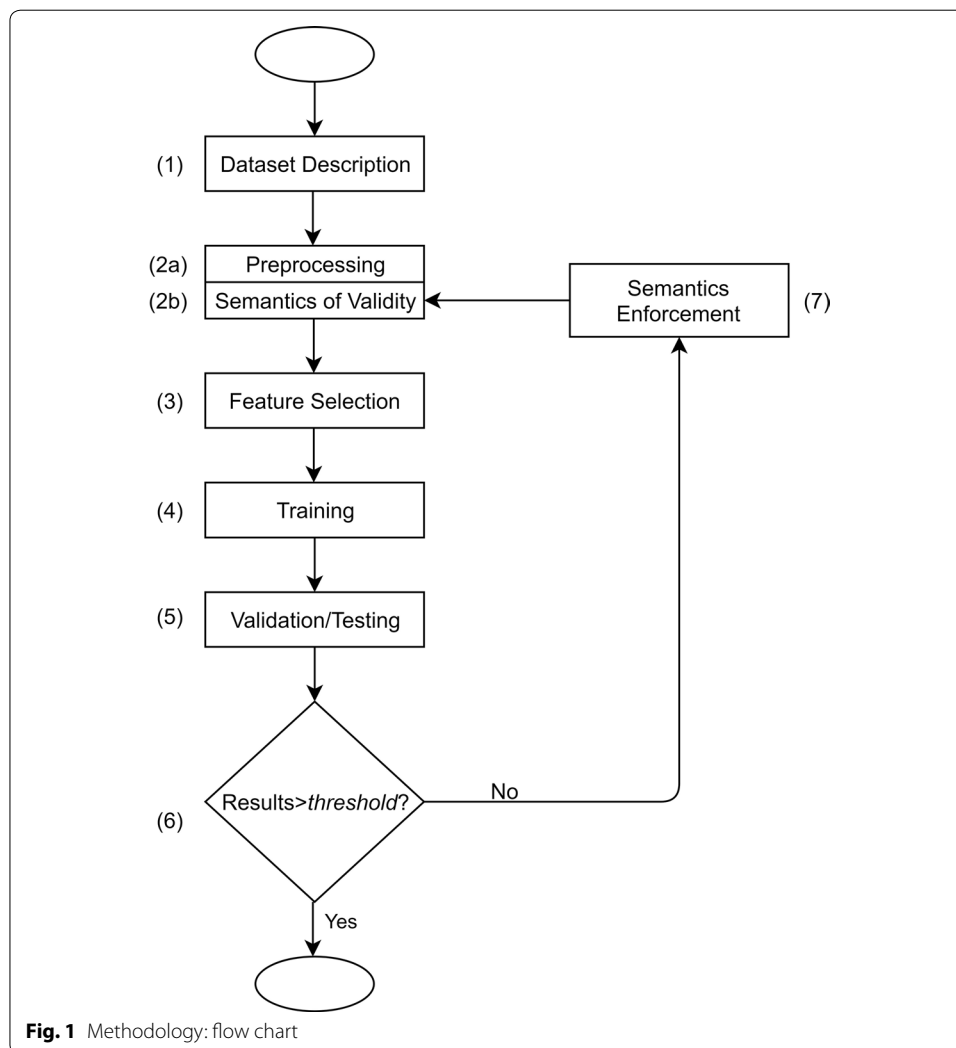
This section presents details on the methodology we have adopted to approach the very complex dataset we were provided with, at the initial stage of our study. Precisely, it has gone through several different phases, that are summarized in Fig. 1.

We here anticipate the meaning of those phases, each of which will be discussed at length in the next subsections.

After an initial description of the dataset (phase 1), preprocessing activities (phases 2) take place. Here a special notice is in order. In our particular case, preprocessing activities are split over two different subphases. The first preprocessing subphase amounts to standard procedure (phase 2a), like encoding and standardization, while the second subphase aims at individuating a semantics of validity for our data (phase 2b), with the hope of filtering out all the impurities that those data carry. (We will come back to this important issue regarding data semantics later on.) After those phases, features are selected (phase 3) to train an intelligent classifier (with the aim of predicting when a water meter fails/needs disassembly). Upon completion of the training activity (phase 4), a validation/testing phase is conducted to assess the performance of the classifier (phase 5). Different from other similar schemes, our methodology incorporates a final verification (phase 6) of the accuracy got so far. We have resorted here to what the literature indicates [17]. A classifier can be: (i) excellent, (ii) good, (iii) fair, (iv) poor, depending on the level of the accuracy it can get, while making predictions. In particular, *excellent* gets a 90–99% of accuracy, *good* gets 80–89%, *fair* gets 70–79%, *poor* 50–69%.

A classifier of reasonable quality cannot, hence, fall down below the threshold of 75%, according to [17]. Based on this consideration, we have taken the results coming from the validation/testing phase and contrasted them against the threshold of 75%. If this control succeeds, we are done. In the negative case, instead, we go back to phase 7.

In the unfortunate case we are sent to phase 7, a new semantics needs to be identified to make the training data suitable for machine learning activities. It is clear that in such a case, more restrictive rules are to be applied to select a subset of data with safe characteristics.



After this introduction, it is now time to describe each of these phases in isolation.

Dataset description

We were, initially, provided with a huge dataset comprised of almost 15 million water meter readings, plus other contextual information. This large dataset spanned a period in time, from the beginning of 2014 to the end of 2018. All those measurements involved more than 1 million water meters, including those affected by faults, and hence subjected to disassembly and subsequent replacement activities.

Our dataset had 14 attributes for each water meter reading, as described in Table 1. Instead, water meters were characterized by 17 attributes, reported in Table 2. It is obvious that negative examples should be faulty meters (with their corresponding readings) and positive examples non-faulty meters (with their corresponding readings). Such information is reported in the attribute *Operation (Faulty/Non Faulty)* of the water meter dataset, which essentially indicates if the water meter has been either disassembled or not.

Table 1 Reading dataset attributes

| No | Attribute name | No | Attribute name |
|----|------------------------|----|--------------------------|
| 1 | Water Meter ID | 8 | Reader ID |
| 2 | Reading ID | 9 | Type of Contract |
| 3 | Reading Value | 10 | Reading Validity |
| 4 | Reading Date | 11 | Certification on the ERP |
| 5 | Previous Reading Value | 12 | Final Billing |
| 6 | Previous Reading Date | 13 | Reason for Reading |
| 7 | Reading Frequency | 14 | Accessibility |

Table 2 Water meter dataset attributes

| No | Attribute name | No | Attribute name |
|----|-------------------------------|----|-------------------------------|
| 1 | Water meter ID | 10 | Installation Date |
| 2 | Serial Number of the Producer | 11 | Plant |
| 3 | Producer Description | 12 | Type of Contract |
| 4 | Material ID | 13 | Geographical Zone |
| 5 | Material Description | 14 | Accessibility |
| 6 | Max/min Reading Value | 15 | Use Category |
| 7 | Meter Type ID | 16 | Address |
| 8 | Meter Type Description | 17 | Operation (Faulty/Non faulty) |
| 9 | Year of Construction | | |

Dataset preprocessing and semantics of validity

This typical phase of preparation of data for machine learning algorithms could almost go without any specific explanation. It simply amounts to the transformation of all the categorical data into numerical ones. We have carried out this phase with the so-called *One Hot Encoding* method, and then we have standardized all the numerical values by subtracting the mean and dividing for the standard deviation, according to the well-known formula: $z = \frac{x-\mu}{\sigma}$.

More interesting, instead, is here the identification of a semantics of validity for the provided readings. In fact, many of the readings that compose the datasets came with numerous *inconsistencies* and *impurities*, whose causes trace down to the point where different business processes had organizational conflicts that are too complex to be explained in detail.

In order to define a semantics of data validity, we took advantage of domain experts. A first step towards a semantics of data validity was taken considering the reading attribute #10 (Reading Validity). This corresponds to the case when a human operator reads a value on a water meter and validates it as correct. In the absence of such a positive validation, that reading is to be considered as non-valid, and should not be taken into consideration. Table 3 reports the number of non-valid measurements with respect to the total amount of circa 15 million readings.

Not only the attribute #10 contributes to the validity of the data, but also attributes #11 (Certification on the ERP) and #12 (Final Billing) play an important role. In fact, if

Table 3 Readings: valid/non-valid (attribute #10)

| Attribute #10 | # of readings |
|---------------|---------------|
| Initial | 15,129,379 |
| Non-valid | 1,898,128 |
| Valid | 13,231,251 |

Table 4 Readings: main categories (with relative amount of readings)

| Attributes | | | # of readings |
|------------|-----|-----|---------------|
| #10 | #11 | #12 | |
| 1 | 2 | 2 | 11,856,582 |
| 1 | 3 | 2 | 407,592 |
| 1 | 2 | 4 | 282,527 |
| 1 | 2 | 6 | 132,409 |
| 1 | 2 | 5 | 110,363 |
| 1 | 2 | 3 | 106,742 |
| 1 | 3 | 5 | 105,957 |
| Other | | | 229,079 |
| Total | | | 13,231,251 |

we join attributes #10, #11, and #12, we yield the following semantics, as suggested by company experts. Consequently, a reading becomes valid if it:

1. has been (correctly) read/collected on site by a human operator,
2. has been (correctly) recorded onto the company ERP system,
3. has been (correctly) billed to the final client.

There would be a total of 45 different combinations that attributes #10, #11, and #12 can take. However, just 7 of those combinations cover almost 99% of the total amount of readings in the dataset, as shown in the first seven lines in Table 4. What is important to know now is that the company assigns to each of these combinations of attributes a given degree of reliability, in terms of data validity. The company, precisely, considers as *fully valid* only the one within the top position in Table 4 (codes: #10 = 1, #11 = #12 = 2). From now on, for the sake of simplicity, we will refer to those readings as those enjoying the 1-2-2 Factor.

At this point, we are interested in understanding how many examples can be assembled to instruct a learning machine. To this aim, we have preliminarily counted how many meters are comprised in the initial dataset, that have a reading history with respectively: at least 1, at least 2, at least 3, at least 4, at least 5 readings, all with the 1-2-2 Factor. The result is shown in Table 5 below, where we have counted both: faulty (line 2) and non-faulty meters (lines 3–7), with their corresponding amounts of readings. The table also reports the total number of water meters (first line).

Unfortunately, of the total amount of readings considered valid by the company (i.e., those enjoying the 1-2-2 Factor) many of them are not real measurements taken on the field by reading a water meter. Indeed, they are mathematical re-adjustments

Table 5 Meters (with the 1-2-2 Factor)

| 1-2-2 Factor | # of meters |
|---------------------|-------------|
| Total | 1,239,977 |
| Faulty (≥ 1) | 23,752 |
| 1-2-2 (≥ 1) | 1,154,054 |
| 1-2-2 (≥ 2) | 1,091,334 |
| 1-2-2 (≥ 3) | 1,038,337 |
| 1-2-2 (≥ 4) | 981,420 |
| 1-2-2 (≥ 5) | 915,441 |

Table 6 Proportion of real measurements vs adjustments (with the 1-2-2 Factor)

| Factor | | | # of readings |
|-------------|---------|---------|-----------------|
| #10 = 1 | #11 = 2 | #12 = 2 | |
| Real | | | 8,185,163 (69%) |
| Adjustments | | | 3,671,419 (31%) |
| Total | | | 11,856,582 |

of estimated values of presumed water consumption values (computed for billing purposes). The balance between real measurements vs these re-adjustments is shown in Table 6. Obviously, such re-adjustments bring much noise and are to be removed consequently.

Feature selection

Now is the time to choose features to be used in the training phase. This is a key task, since irrelevant or redundant features can impact the training activities [18, 19]. In our case, nonetheless, this thing goes smooth as reading values (current and previous), and relative dates, compose the minimal set of information from which learning algorithms can extract interesting relationships (attributes #3, #4, #5, #6 of the readings dataset). Further, on the basis of precise suggestions provided by the company, we also included the following additional features from the meter datasets: producer (attribute #2), material (attribute #4), meter type (attribute #7), year of construction (attribute #9), and use category (attribute #15). Summarizing, Table 7 reports all the aforementioned selected features.

Training, validation, and testing

A deep neural network was used for training. Given the nature of the data, where also the passage of time has a value (some few consecutive readings are used to make a decision), we developed a recurrent neural model suitable for managing time, using the Keras framework. In particular, our neural network is comprised of two parallel subnets.

Take the first one. It is intended to learn series of consecutive readings. It presents a Gated Recurrent Unit (GRU) for each reading in the series. The output of each GRU is passed to a Dense layer of 32 fully connected neurons. The overfitting phenomenon

Table 7 Features used

| No | Features |
|----|-------------------------------|
| 1 | Reading Value |
| 2 | Previous Reading Value |
| 3 | Reading Date |
| 4 | Previous Reading Date |
| 5 | Serial Number of the Producer |
| 6 | Material ID |
| 7 | Meter Type ID |
| 8 | Year of Construction |
| 9 | Use Category |

is avoided by using an in-between Dropout layer, with a keep probability of 0.9, that separates GRUs from the Dense layer.

Consider now the second subnet. It takes as input the one-hot encoded categorical features we have selected and lets them pass through two Dense layers of fully interconnected neurons. The first layer has 128 neurons, while the second one has just 32 neurons. Again, an in-between Dropout layer, with a keep probability of 0.9, separates the two layers of neurons to avoid the overfitting phenomenon.

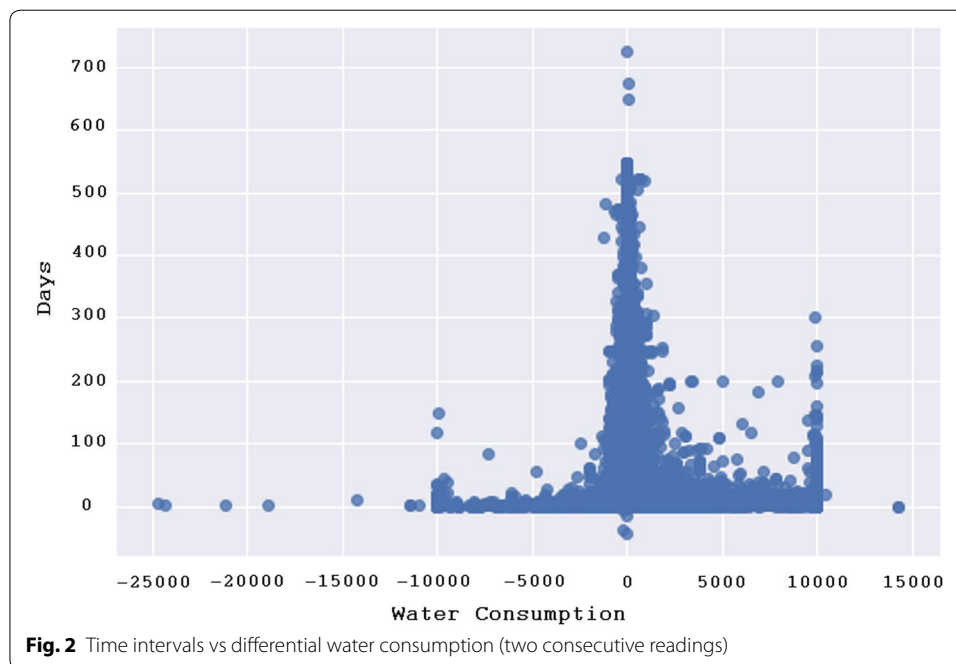
At this stage, the two parallel outputs of the two subnets are concatenated to form a 64-dimensional vector that passes through a further Dense layer comprised of 64 neurons. The output of this step is a two-dimensional vector (faulty/non faulty) which is finally delivered to a Softmax activation function that yields the final probability of being faulty or not.

It is worth to notice also that each mentioned layer uses a REctified Linear Unit (RELU) as activation function, while we employed a Binary Cross-Entropy function to manage losses, based on the consideration that we had to construct a binary classifier. To conclude the description of our network, we add that it was trained using the well-known Gradient Descent Algorithm, for 20 epochs, to yield the final optimization.

At this point, we made a first attempt to train our deep neural network with negative/positive examples that were assembled by randomly sampling faulty and non-faulty meters with their corresponding readings enjoying the 1-2-2 Factor. We conducted this training activity with series of readings of different lengths, containing either two or three consecutive readings, taken in the period beginning 2014–mid 2018. We used the well-known *tenfold cross-validation* technique, where a portion of the data is used for *training* and the remaining one for *validation*.

As these two classes, faulty and non-faulty, in our case, are highly imbalanced (see Table 5 before), we are in the presence of a typical unbalanced training problem, with risk of a bias in favor of the majority class. To fix this problem, we employed the well-known SMOTE-NC technique [20, 21] that oversamples the quantity of faulty water meters, until the cardinality of the two classes in the training set becomes equal.

To conclude, we come to the evaluation metric we have adopted to measure the accuracy of our classifier. We have chosen the classic *area under the curve* of the *receiver operating characteristic* (AUC-ROC) [22].



As mentioned at the beginning of “[Methodology](#)” section, it is worth reminding that we get a good classifier only if it achieves an AUC ROC value greater than 0.75. Now, the problem with the process we have conducted so far is that if we validate our neural model, trained with data enjoying the 1-2-2 Factor, we never surpass AUC ROC accuracy values of 0.61. Such negative result denotes a problem in the semantics of data validity that needs to be fixed, as discussed to the next subsection.

Semantics enforcement

What the negative accuracy result mentioned in the previous subsection has emphasized is that the 1-2-2 Factor data semantics, suggested by the company, is not sufficient to guarantee a safe training activity for the learning algorithms.

Our intuition is that that semantics disregards the role played by time. In essence, of great importance is the time interval between two consecutive valid readings, taken on a given meter. In fact, to use those data to train a neural network, crucial is the regularity of the frequency with which a reading, with the 1-2-2 Factor, is read over time.

Figure 2 provides insightful information with this regard. On the *x-axis* of Fig. 2, plotted are the differences of the two values (i.e., in terms of cubic meters of consumed water) recorded at two different subsequent readings enjoying the 1-2-2 Factor, while on the *y-axis* we can see the time intervals (measured in days) between two subsequent readings with the 1-2-2 Factor.

This Fig. 2 summarizes some millions of reading values taken over a lot of time, and it has to be interpreted as follows. Points, that lies on the *y-axis* and are very far from zero, correspond to measurements that are not taken without any regularity (while, instead, Italian laws prescribe two/three real readings per year). Points, that lies on the *x-axis* and are very far from zero, account instead for phenomena where the consumption of water is exaggeratedly high.

At the end, all this seriously questions the validity of using water meter readings enjoying the 1-2-2 Factor for training an intelligent classifier, even though they are those readings considered as the most reliable ones by the company.

Summing up, looking at this problem from all the possible perspectives, the data that were made available to us in their initial forms cannot be considered a good starting point to train a machine, as they can hardly provide unambiguous examples to be learnt by a learning algorithm.

To make the complex piles of information described above genuinely valid for carrying out learning activities, we then have to move towards an alternative approach, at the basis of which lies a procedure developed to clean the data, while reducing their dimension.

In essence, a new *semantics of validity* has to be defined for our readings that can be summarized as follows. A reading is to be considered valid only if all the following requirements are satisfied:

1. a human operator has read a certain reading value at the reading site;
2. that reading value has been correctly recorded onto the company ERP and billed to the client; and most important,
3. the instants in time when that reading was subjected to previous actions 1 and 2 are temporally valid values, and as such certified by a specific business process.

Said simpler, this new semantics confirms the 1-2-2 Factor which valid readings must possess, plus a certification on the validity of the temporal dates when a given reading is read and then collected by a human operator. Indeed, the rationale behind the enforcement of this third requirement was to admit as valid only those reading values that are not too far each from other, from a temporal viewpoint.

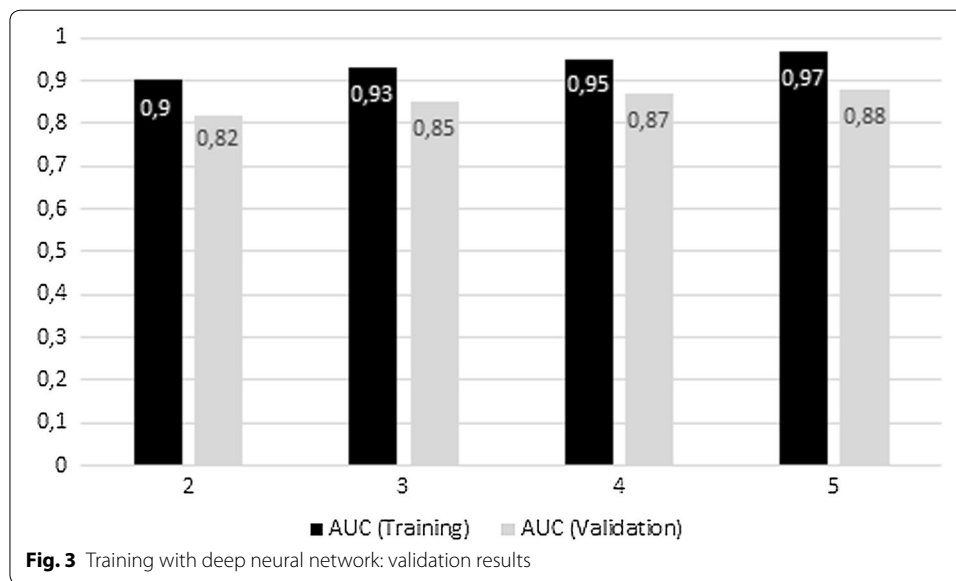
For the sake of simplicity, we will call this enhanced semantics, from now on, as the X-Factor. Before closing this subsection, of paramount importance is to notice that the enforcement of the X-Factor to our initial dataset makes the number of valid readings falling down to less than two million. On this reduced number of readings, we will instruct our machines, as described in the following section.

Results

This section goes through three different phases. First, we present the accuracy results of the prediction we have obtained using training data enjoying the X-Factor. Second, we provided a comparative analysis of the results we have obtained with our neural network contrasted against those that can be obtained with alternative, more traditional methods. Finally, in the third part, we discuss on a statistical paradox that the transformation of the initial dataset has brought with it. These three facts are discussed in the remainder of this section, in isolation.

Testing results

We carried out our machine training activities with meters (faulty and non-faulty) whose readings enjoyed the X-Factor, always taken in the period beginning 2014–mid 2018. We assembled a set of positive examples comprised of some 45,000 non-faulty meters with



all readings enjoying the X-Factor, along with a set of negative examples comprised of some 15,000 faulty meters (with correspondent readings enjoying the X-Factor). Using SMOTE-NC, we oversampled the faulty water meters, until reaching the amount of 45,000. We used again the deep neural network, whose description is reported in the subsection termed “[Training, validation, and testing](#)”. We experimented with series of readings of different lengths, to take advantage of the memory of the network.

The results of the cross-validation are reported in Fig. 3. Of particular interest are the average validation results (gray bars) that provide accuracy values always over the threshold of 80%, precisely in the range [82–88] %.

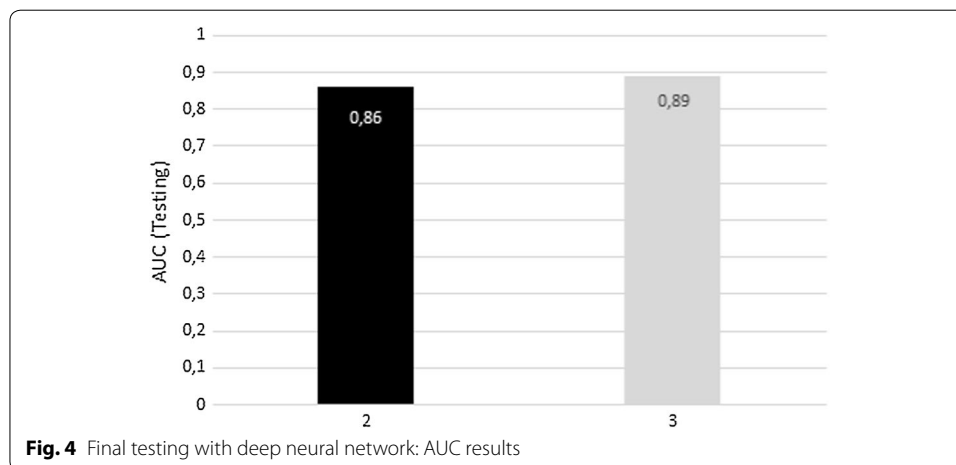
These preliminary results obtained during the validation phase of our neural network training process were well promising, yet we wanted to have a final confirmation of the efficacy of the combination of a deep neural network plus the X-Factor. To this aim, we developed an additional, and final, testing experiment of our deep neural network, using only that portion of meters (precisely 29,286), with readings enjoying the X-Factor, that our machine never looked at during the training phase. Actually, all those readings with the X-Factor recorded in the period mid 2018–end 2018.

Just series of either two or three readings were tested, for the sake of simplicity. Figure 4 portrays those results. A confirmation of the efficacy of our combo (neural network + X-Factor) comes under the form of an accuracy in the range [86–89] %, depending on the number of consecutive readings exploited to make the decision.

Comparative analysis

We have conducted a comparative analysis to compare the performance of our deep neural network with some of the most common machine learning algorithms that, different from our recurrent deep network, do not use memory.

In this case, we employed approx. 15,000 faulty meters, with their correspondent valid readings (approx. 80,000) and a set of almost 100,000 non-faulty meters (with



their correspondent valid readings). We experimented with all the following traditional learning algorithms:

- Linear Regression (LR),
- Lasso (LA),
- Elastic Net (EN),
- Classification and Regression Tree (CART),
- Support Vector Regression (SVR),
- K-nearest neighbors (KNN),
- Adaptive Boosting (AB),
- Gradient Boosting (GB),
- Random Forest (RF),
- Multi-Layer Perceptron (MLP with only one hidden layer with 100 neurons).

In Fig. 5, a plot with the accuracy results obtained during validation with each aforementioned algorithm is reported, contrasted against the result achieved with our deep neural network (DNN) with three readings.

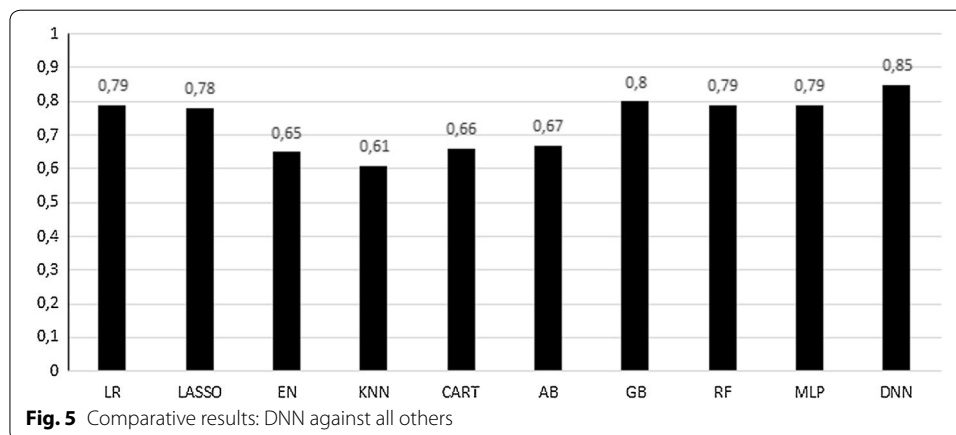
The figure shows that acceptable average AUC ROC values can be achieved in some specific case; for example, with the GB and MLP algorithms, that yield almost an 80% value of accuracy. Nonetheless, DNN performs significantly better as it reaches an average accuracy value of 85% with just three readings.

A statistical paradox

At this point, satisfied with the precision on the level of prediction accuracy reached upon enforcing the so-called X-Factor to our training data, we began to reflect on the statistical meaning and validity of the operations we had carried out so far.

We simply asked ourselves: *Did the X-Factor semantics just re-adjusted our data by cleaning them from the initial impurities, or it actually transformed them from some (statistical) viewpoint?*

To answer this question, we conducted various different statistical tests.

**Table 8** Statistics

| Id | Dataset | # of readings | μ | σ |
|----|-------------------------|---------------|-------|----------|
| 1 | Total with 1-2-2 Factor | 11,856,582 | 5307 | 86,450 |
| 2 | Total with X-Factor | 1,973,493 | 3674 | 17,796 |
| 3 | Sampled for training | 135,018 | 3647 | 11,852 |

We start this analysis with Table 8 where reported are, respectively, the total number of readings: (1) with the 1-2-2 Factor, (2) with the X-Factor, and (3) belonging to the subset that was used for training of our neural network, with a series of three consecutive readings enjoying the X-Factor. Table 8 also reports the average (μ) and the standard deviation (σ) values of the consumed water per reading (cubic meters of consumed water).

We were surprised by a fact in that table. The average consumption of water per reading falls down to approx. 3.000 m³ when the X-Factor is applied. Consequently, we developed some statistical tests aimed at better understanding this fact.

We first started by assuming a normal distribution (with known values for both average and standard deviation) and proceeded with a *Z Test*, whose results are reported in Table 9. We tested our null hypotheses with two different *significance* α values. As seen from Table 9, (not) surprisingly the null hypothesis that the average values of consumed water per reading in the initial dataset and in the subset of readings subjected to the X-Factor are equal is to be rejected (first line). Instead, as expected, fortunately, it cannot be rejected the null hypothesis that the whole subset of readings with the X-Factor has an average value of consumed water per reading equal to the specific subset of those readings specifically used for training.

To have a further confirmation, we repeated the same kind of test, yet with a different statistic. Simply, we tried to use a *Student's T test* (assuming not to know the standard deviation values). This has to be intended just as an additional attempt to verify the previous results and, in fact, not surprisingly, we got very similar outcomes, as Table 10 demonstrates.

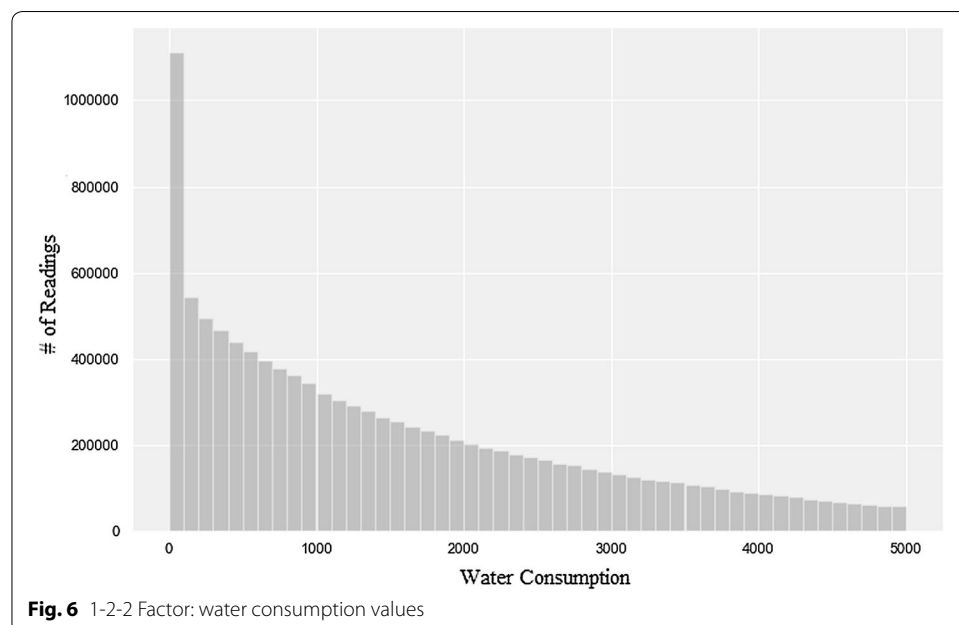
Nonetheless, if we look at this issue from a different perspective, we can observe some aspects that bring us into a more comfortable zone. Take into consideration, for

Table 9 Z test—results

| Test | p-value | $\alpha = 0.05$ | $\alpha = 0.01$ |
|-----------------|-------------|-----------------|-----------------|
| $\mu_1 = \mu_2$ | $< 10^{-5}$ | Reject | Reject |
| $\mu_2 = \mu_3$ | 0.75 | Fail to reject | Fail to reject |
| $\mu_1 = \mu_3$ | $< 10^{-5}$ | Reject | Reject |

Table 10 T test—results

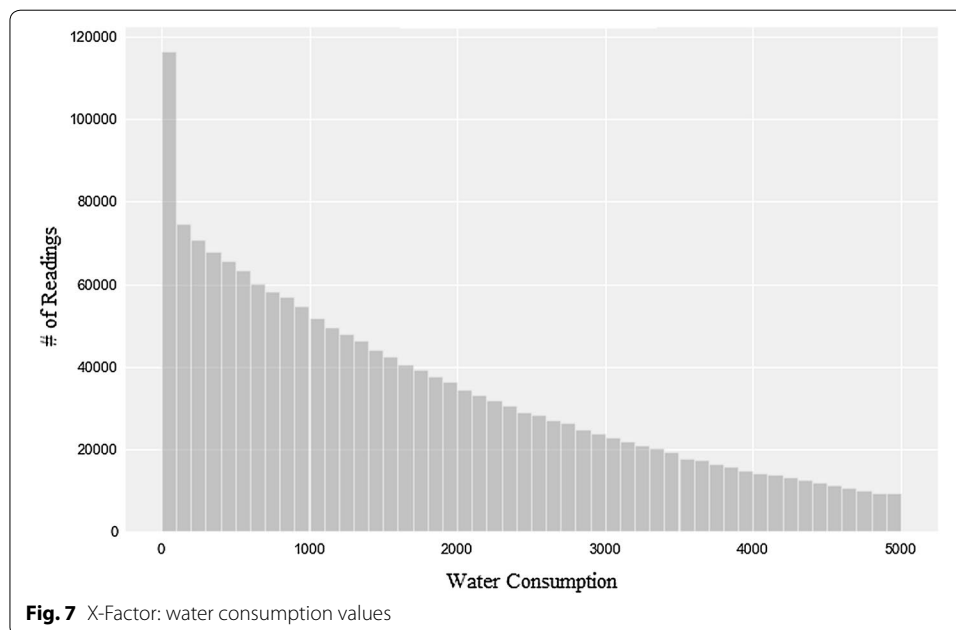
| Test | p-value | $\alpha = 0.05$ | $\alpha = 0.01$ |
|-----------------|-------------|-----------------|-----------------|
| $\mu_1 = \mu_2$ | $< 10^{-5}$ | Reject | Reject |
| $\mu_2 = \mu_3$ | 0.62 | Fail to reject | Fail to reject |
| $\mu_1 = \mu_3$ | $< 10^{-5}$ | Reject | Reject |



example, the plots of Figs. 6 and 7. They both aim to measure the number of readings (y axis) whose average value equals a given value, say X (on the x axis). In Fig. 6 we have the case of the dataset with the 1-2-2 Factor, while in Fig. 7 we have the X-Factor case.

As seen from a visual comparison of the two plots, we have a clear impression that the shapes of the two curves are not that different, even if the quantity of readings with an amount of consumed water equal to any given value X in the first dataset is larger than the correspondent quantity of readings with the X-Factor, in this sense confirming the results of the statistical tests of Tables 9 and 10.

To finally understand, we developed some additional statistical tests more focused on the shapes of the distributions of the average value of consumed water for respectively: the dataset with the 1-2-2 Factor (1), all the readings with the X-Factor (2) and just that subset of readings with the X-Factor used for training our neural network (3). Being our

**Table 11** Kolmogorov–Smirnov Test—results

| Test | p-value | $\alpha = 0.05$ | $\alpha = 0.01$ |
|----------|-------------|-----------------|-----------------|
| KS (1,2) | $< 10^{-5}$ | Reject | Reject |
| KS (2,3) | 0.24 | Fail to reject | Fail to reject |
| KS (1,3) | $< 10^{-5}$ | Reject | Reject |

Table 12 Chi-squared test—results

| Test | p-value | alpha = 0.05 | alpha = 0.01 |
|-------------|-------------|----------------|----------------|
| X^2 (1,2) | $< 10^{-5}$ | Reject | Reject |
| X^2 (2,3) | 0.125 | Fail to reject | Fail to reject |
| X^2 (1,3) | $< 10^{-5}$ | Reject | Reject |

variables numerical, we first used a *Kolmogorov–Smirnov Test*, with two different *significance* α values. Table 11 portrays the results that confirm what was already clear from Tables 9 and 10. Even the general distributions are different if we consider readings with the 1-2-2 Factor and those with just the X-Factor.

This story does not change even if we try with a different statistical test. We discretized our data, clustered the results into bins and then went for a *Chi-squared Test*. Results are shown in Table 12. The results do not change.

Before going, with the next section, towards a direction where this paradox can be reconciled, a consideration is in order to conclude this present section: *As we aimed to improve our machine learning performance results in terms of accuracy of prediction, enabling the transformation of data through re-organization, we simultaneously changed the replicated forms of those data. At least, statistically.*

Discussion

If we take into serious consideration both the statistical paradox we have illustrated in the previous section, and the positive prediction outcomes obtained with the classifier we have trained with just those data enjoying the X-Factor, a logical consequence follows: *Bigger is no longer better, when big also implies confuse, inconsistent and biased information*. Nonetheless, and not simplistically, all our effort towards the definition and the enforcement of the X-Factor semantics has taught us two fundamental facts:

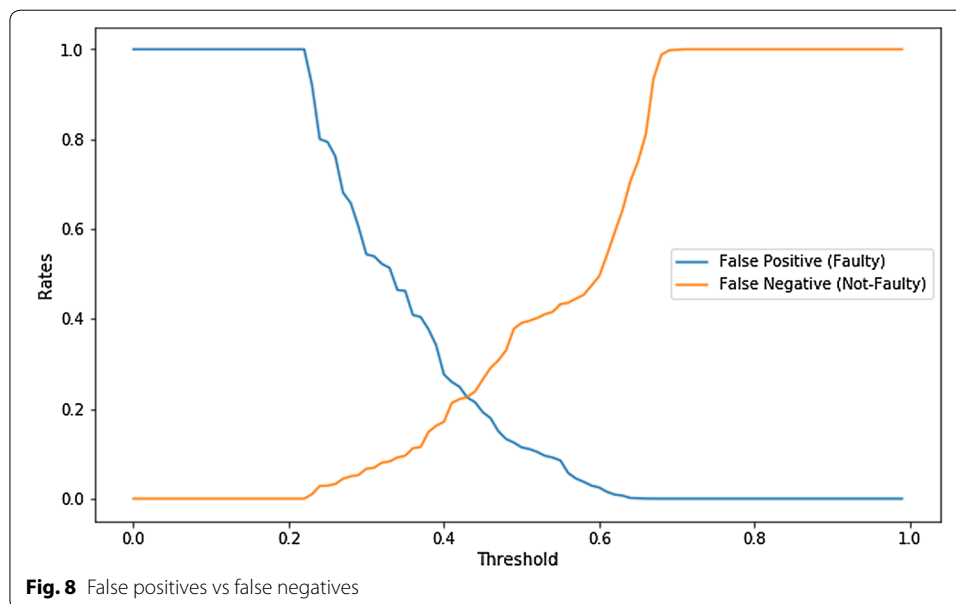
- Not all the collected information that serves the interests of some specific business process can be considered adequate to instruct an intelligent machine that is intended to implement a new service, if this process is conducted without a deep reflection on the validity, sense and subtle implications of the training data;
- Nonetheless, that kind of information has not to be deleted or radically transformed, until it serves profitably the interests of those organizational processes, on which the company has traditionally based its business. In our case, setting a goal of transformation of all the relevant business processes only to obtain those readings values with the X-Factor that are adequate for training an intelligent machine is both unrealistic and nonsensical.

Thus, a final question is in order: All this considered, how the company described in our study could take advantage of our intelligent classifier, that works well only in the case it examines water meter reading values with the X-Factor?

To help with this question, comes the narration of the procedure that the company currently exploits to detect faulty meters.

It is based, first, on a software procedure that considers a meter as a candidate to be faulty if two consecutive readings (with the 1-2-2 Factor) are read with almost a null increment in water consumption. These are only *candidates*, though. Then an expensive, human-based process starts to individuate, among the candidates, those meters that actually need a replacement. To understand how much complex and expensive is such a final process, consider that in many cases verification/repair interventions are scheduled and performed by human operators, who have to reach the place where the meter is installed. In other cases, controls are performed to make a report of suspected failure, by screening additional databases containing relevant information that could validate or not that suspect. For example, it could be the case when both water and gas are supplied by the same company. In this case, to confirm a suspect of a failure occurring at a given water meter, the gas meter serving the same client should be recording a non-null gas consumption. Finally, note also that not all those meters that are *greatly suspected* as faulty are finally changed, due to both operational and business motivations, whose discussion is out of the scope of this paper.

It is important, at this point, to talk about a little of statistics regarding the company of interest. On average, per year: some 10,000 m are considered as faulty *candidates*, based on the estimates the company makes. Always on average, almost 5500 are those meters that are *greatly suspected* as faulty after the execution of the complex procedures mentioned above, while some 1.500 m are finally *replaced* in a year. The reader should put attention to this latter value of 1500 replaced meters per year, as this is



currently the *maximum amount* of faulty meters that the company can replace, based on its replacement policy.

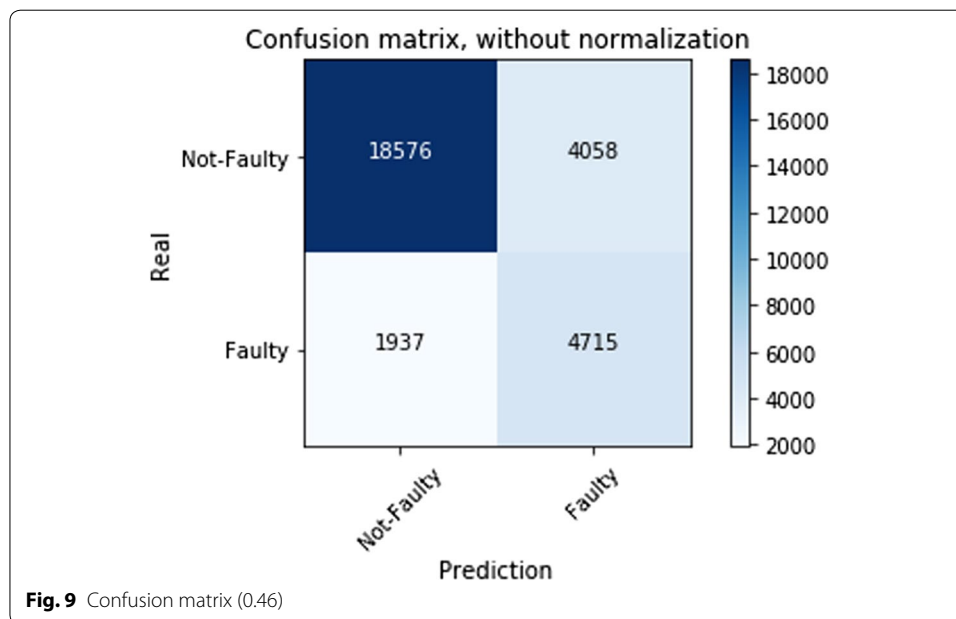
If we take into consideration these statistics, all our previous results resurface now under a new guise.

Take for example the results we got with the testing phase conducted with almost 30,000 m, with readings enjoying the X-Factor, in the period mid 2018–end 2018 (as described in the previous “Results” section). Out of those 30,000 m, 6652 m were *suspected* as faulty, while 22,634 were the non-faulty ones, as stated by the company. Just to remind it, our classifier was able to make predictions in that context with an accuracy of 86%, in terms of the AUC-ROC metric (in the case two X-Factor readings were used).

We then could initially try to use our classifier, set with a decision threshold of 0.46, the one that minimizes the number of both the false negative (faulty meters predicted as non-faulty) and the false positive (non-faulty meters predicted as faulty), as per Fig. 8. With that threshold, we would obtain a classification of faulty/non-faulty, like that represented in the confusion matrix of Fig. 9.

At this stage, we could propose two alternative operational approaches to replace faulty meters, that combine the results of our classifier with the traditional procedures already in use.

With the first one, we could suggest to the company not to scrutinize all the meters that our classifier has predicted as non-faulty (precisely 20,513 m, computed as the sum of the top and the bottom quantities on the left side of Fig. 9), and to concentrate their attention, as well as to deploy their traditional verification procedures, only to those meters that were predicted as faulty (i.e., *greatly suspected*; precisely 8773 m, obtained by summing all the quantities on the right side of Fig. 9). Unfortunately, this approach does not work well in this case, due to the fact that the company should use its traditional and expensive verification procedures on a number of meters (8773, for a 6-months-long



period) that almost doubles the average quantity of meters that are considered as faulty with the methods already in use (5500, over a period of a year).

Not only, with this approach we know for sure that some 1937 faulty meters will never be detected (left bottom sector in Fig. 9).

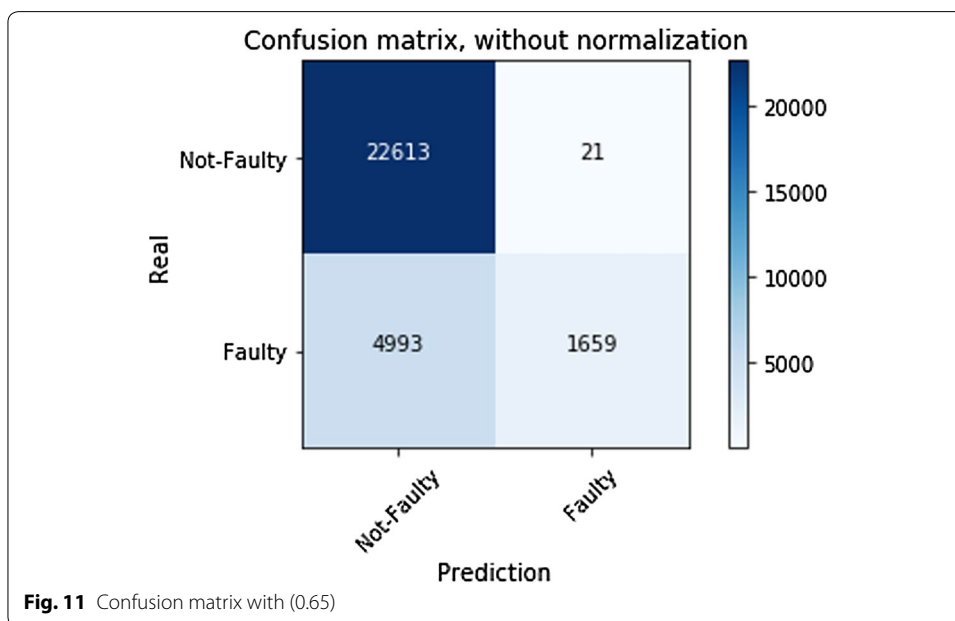
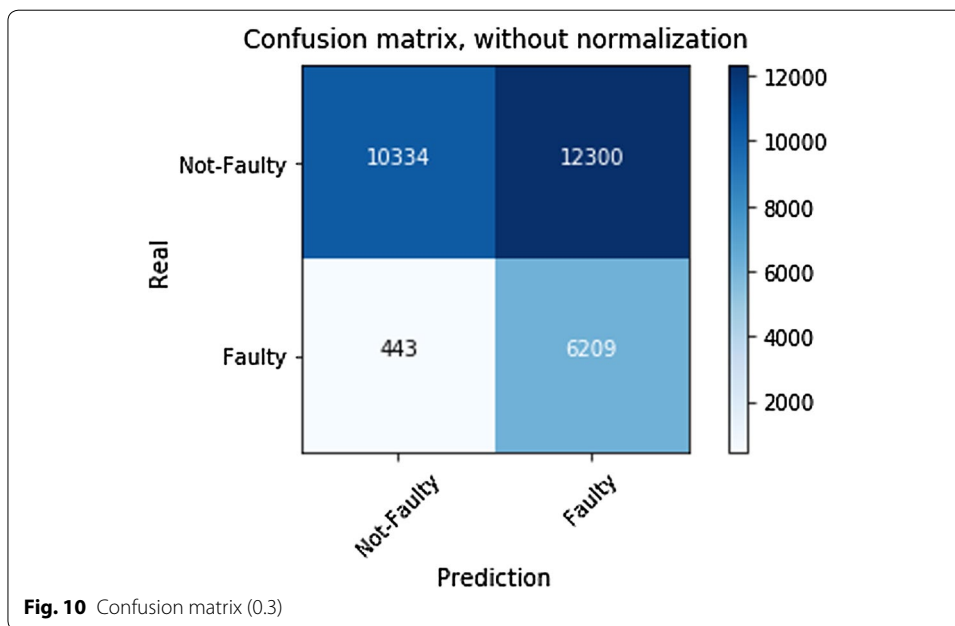
To solve this latter problem, we could then move the decision threshold towards the direction of minimizing the number of false negatives [23]; for example, setting the decision threshold at the value of 0.3, as per the confusion matrix of Fig. 10. This would have the effect of decreasing the number of faulty meters that are never detected down to 443 (left bottom sector in Fig. 10). Unfortunately, this way, we have further exacerbated the problem of scrutinizing a huge number of meters that are suspected of being faulty, yielding almost 18,509 water meters to be verified with expensive procedures (computed as the sum of the quantities on the right side of Fig. 10).

Well promising, instead, is the second approach we propose.

The idea is that of minimizing the number of false positives, for example moving the decision threshold to a value of 0.65, as per Fig. 11. With this approach, our suggestion to the company is to adopt a brand, new procedure to replace faulty meters, which is as follows: Put the focus only on the meters that our classifier has predicted as faulty (1680 m on the right side of Fig. 11), and proceed directly with the replacement of all those meters.

Following this approach, the company will have to replace a number of faulty meters which is comparable to the maximum number of those it can replace based on its current replacement policy (1680 vs circa 1500), yet without the need to resort to complex and expensive procedures to individuate them.

Further, in this situation minimized is also the amount of those meters that go replaced even if they did not need any replacement. Indeed, only 21. In other words, in this case, meters have been predicted as faulty and then replaced with a precision of 98.75%. As a



final consideration, this discussion demonstrates how a savvy use of our intelligent classifier can help the company to detect faulty meters to be replaced without any interference on the business process currently in use [24].

Anyway, to conclude this discussion it is important to underline the very general consideration that, even though our classifier has been trained only with readings enjoying the so-called X-Factor, it can be used to predict failures for each and any meter run by the company, subject to the (simple) condition that that meter provides at least two or three valid readings per year (i.e., with the X-Factor).

Conclusions

We have extrapolated from a database of almost fifteen million water meter readings just those data that could safely represent a complex phenomenon of water consumption leading to some meter failures. We have re-modeled the initial dataset of water meter readings, based on a new data semantics (termed the X-Factor). On one side, this has allowed us to design a ML-based classifier able to predict if a meter has failed/needs a replacement, based on a history of water consumption measurements, that yields accuracy values over the threshold of 80%. On the other side, we have become aware that the data on which we have trained our machine have some statistical *discrepancies* with respect to those comprised in the initial dataset, thus reaching a kind of apparent paradox. We have reconciled this paradox, showing how an adequate use of our classifier can help the company that provided the initial data to detect the meters to be replaced, at a lower cost than that previously paid when different and more expensive procedures were in use. This completes our controversial journey that began almost a year ago and of which some very preliminary studies can be also found in [25, 26].

Abbreviations

AI: artificial intelligence; ML: machine learning; AUC: area under the curve; ROC: receiver operating characteristic; GRU: Gated Recurrent Unit; RELU: REctified Linear Unit.

Acknowledgements

We are indebted towards the company that has provided us with the data of interest. To guarantee its privacy, we keep it here anonymized. We are also grateful to all the reviewers who helped us to improve the quality of the paper. Finally, we thank Prof G. Marfia and Dott. N. Zagni (University of Bologna) for their many helpful suggestions on a previous version of this manuscript.

Authors' contributions

All the authors contributed equally to this manuscript. All authors read and approved the final manuscript.

Authors' information

Marco Rocchetti is Full Professor of Computer Science at the University of Bologna (Italy). He was a former Director of the Ph.D. Program on Data Science and Computation at the University of Bologna. He was also a Visiting Scholar at the University of California Los Angeles and a Visiting Scientist at the International Computer Science Institute in Berkeley. His current research interests include: human-machine-bigdata interaction and human-in-the-loop methods.

Giovanni Delnevo is pursuing a Doctorate Degree on Data Science and Computation at the University of Bologna. His main research interests include: machine learning and human machine interaction.

Luca Casini is a Ph.D. Candidate at the Data Science and Computation Program at the University of Bologna. His main research interests include: deep Learning for creative tasks and issues of human-AI interaction.

Giuseppe Cappiello is an Assistant Professor at the Department of Management of the University of Bologna and was also a Visiting Scholar at the Kellogg Graduate School of Management, Evanston (IL). His research interest touches upon issues of technological innovation in companies and business processes.

Funding

Not applicable.

Availability of data and materials

The datasets used and/or analyzed during the current study are available from the corresponding author on reasonable request.

Competing interests

The authors declare that they have no competing interests.

Author details

¹ Department of Computer Science and Engineering, University of Bologna, Via Mura Anteo Zamboni 7, 40127 Bologna, Italy. ² Department of Management, University of Bologna, Via Capo di Lucca 34, 40126 Bologna, Italy.

Received: 29 May 2019 Accepted: 24 July 2019

Published online: 03 August 2019

References

- Pettersen L. Why artificial intelligence will not outsmart complex knowledge work. *Work, employment and Society*. Thousand Oaks: Sage Pub; 2018 (**in Press**).
- Jordan MJ, Mitchell TM. Machine learning: trends, perspectives, and prospects. *Science*. 2015;349(6245):255–60.
- Delnevo G, Roccetti M, Mirri S. Intelligent and good machines? The role of domain and context codification. *Mobile networks and applications*. Amsterdam: Elsevier; 2019 (**in Press**).
- Witten IH, Frank E, Hall MA, Pal CJ. *Data Mining: practical machine learning tools and techniques*. The Morgan Kaufmann series in data management systems. Burlington: Morgan Kaufmann; 2016.
- Alkowiileet W, Alsubaiee S, Carey M, Li C, Ramampiaro H, Sinthong P, Wang X. Enhancing Big Data with semantics: the AsterixDB approach. In: *Proc. of 12th IEEE international conference on semantic computing, IEEE*. 2018. p. 314–5.
- Emani CK, Cullot N, Nicolle C. Understandable big data: a survey. *Comput Sci Rev*. 2015;17:70–81.
- St. Clair AM, Sinha S. State-of-the-technology review on water pipe condition, deterioration and failure rate prediction models! *Urban Water J*. 2012;9(2):85–112.
- Pietrucha-Urbaniak K. Failure prediction in water supply system-current issues. In: *International conference on dependability and complex systems*. Springer. 2015. p. 351–8.
- Alvisi S, Casellato F, Franchini M, Govoni M, Luciani C, Poltronieri F, Riberto G, Stefanelli C, Tortonesi M. Wireless mid-leware solutions for smart water metering. *Sensors*. 2019;19(8):1853.
- Roberts SE, Monks IR. Fault detection of non-residential water meters. In: Weber T, McPhee MJ, Anderssen RS, editors. *MODSIM2015, 21st international congress on modelling and simulation*. Modelling and simulation society of Australia and New Zealand, December 2015, p. 2228–33. ISBN: 978-0-9872143-5-5.
- Monedero I, Biscarri F, Guerrero JL, Roldán M, León C. An approach to detection of tampering in water meters. *Procedia Comput Sci*. 2015;60:413–21.
- Allen GI. Statistical data integration: challenges and opportunities. *Stat Model*. 2017;17(4–5):332–7.
- Buolamwini J, Geburu T. Gender shades: intersectional accuracy disparities in commercial gender classification. In: *Proc. of international conference on fairness, accountability and transparency, JMLR*. 2018. p. 77–91.
- Bolukbasi T, Chang KW, Zou JY, Saligrama V, Kalai AT. Man is to computer programmer as woman is to homemaker? Debiasing word embeddings. In: *Proc. of advances in neural information processing systems, NIPS Foundation*. 2016. p. 4349–57.
- Pennington J, Socher R, Manning C. Glove: global vectors for word representation. In: *Proc. of 2014 conference on empirical methods in natural language processing, Association for Computational Linguistics*. 2014. p. 1532–43.
- Cabitza F, Rasoini R, Gensini GF. Unintended consequences of machine learning in medicine. *J Am Med Assoc*. 2017;318(6):517–8.
- Carter JV, Pan J, Rai SN, Galandiuk S. ROC-ing along: evaluation and interpretation of receiver operating characteristic curves. *Surgery*. 2016;159(6):1638–45.
- Guyon I, Elisseeff A. An introduction to variable and feature selection. *J Mach Learn Res*. 2003;3(Mar):1157–82.
- Li Z, Wang Y. Domain knowledge in predictive maintenance for water pipe failures. In: Chen F, Zhou J, editors. *Human and machine learning*. Berlin: Springer; 2018. p. 437–57.
- Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP. SMOTE: synthetic minority over-sampling technique. *J Artif Intell Res*. 2002;16:321–57.
- Leevy JL, Khoshgoftaar TM, Bauder RA, Seliya N. A survey on addressing high-class imbalance in big data. *J Big Data*. 2018;5(1):42.
- Tharwat A. Classification assessment methods. *Appl Comput Inform*. 2018. <https://doi.org/10.1016/j.aci.2018.08.003>.
- Mirniaharikandehi S, Hollingsworth AB, Patel B, Heidari M, Liu H, Zheng B. Applying a new computer-aided detection scheme generated imaging marker to predict short-term breast cancer risk. *Phys Med Biol*. 2018;63(10):105005.
- Brock V, Khan HU. Big data analytics: does organizational factor matters impact technology acceptance? *J Big Data*. 2017;4(1):21.
- Casini L, Delnevo G, Roccetti M, Zagni N, Cappiello G. Deep water: predicting water meter failures through a human-machine intelligence collaboration. In: *Proc. of international conference on human interaction & emerging technologies*. 2019. Springer. (**To appear**).
- Roccetti M, Zagni N, Delnevo G, Casini L, Cappiello G. A paradox in ML design: less data for a smarter water metering cognification experience. In: *Proc. of GOODTECHS'19. ACM*. 2019. (**To appear**).

Publisher's Note

Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.