Monitoring and Detecting Faults in Wastewater Treatment Plants using Deep Learning

Behrooz Mamandipoor[a], Mahshid Majd[a], Seyedmostafa Sheikhalishahi[a], Claudio Modena[b], Venet Osmani[a1]

[a]*Fondazione Bruno Kessler Research Institute, via Sommarive 18, Trento, Italy*
[b]*E.T.C. Engineering Solutions, Trento, Italy*

Abstract

Wastewater treatment plants use many sensors to control energy consumption and discharge quality. These sensors produce a vast amount of data which can be efficiently monitored with automatic systems. Consequently, several different statistical and learning methods are proposed in literature which can automatically detect the faults. While these methods showed promising results, the nonlinear dynamics and complex interaction of the variables in wastewater data necessitate more powerful methods with higher learning capacities. In response, this study focusses on modelling faults in oxidation and nitrification process. Specifically, this study investigates a method based on Deep Neural Networks (specifically Long Short-Term Memory) compared with statistical and traditional machine learning methods. The network is specifically designed to capture temporal behaviour of sensor data. The proposed method is evaluated on a real-life dataset containing over 5.1 million sensor data points. The method achieves fault detection rate (recall) of over 92%, thus outperforming traditional methods, and enabling timely detection of collective faults.

Keywords: Wastewater Plant Treatment, Fault Detection, Ammonia feedback, Deep Learning, LSTM

Introduction

Water collected from households and industrial plants must be treated before being discharged into rivers or other water bodies. In this respect, Waste Water Treatment Plants (WWTPs) play an essential role in reducing environmental pollution through removing or breaking down pollutants and reclaiming wastewater. However, WWTPs are complex systems that must maintain high performance, despite temporal dynamics, such as daily and seasonal changes or human activity. To safely and optimally operate a WWTP, it is necessary to monitor the treatment process online which is costly and requires specialized equipment. In response, several sensors are used to monitor the WWTPs influents such as ammonia, dissolved oxygen, several nutrients,

[1] Corresponding author: vosmani@fbk.eu

suspended solids, and organic matter. Yet, it is practically impossible to always either deploy perfectly working sensors, have human experts monitor them or redesign sensor placement (Villez et al., 2016). Consequently, an important research direction is to precisely monitor faults in the sensors. Faults can be of different types and occur at different locations, however this work focuses on fault detection in influent sensors, specifically ammonia measurements sensors in the nitrification oxidation tank. As WWTPs generate a large amount of data, a promising solution lies in automatic detection of such faults in the system using machine learning methods and algorithms to automatically process the data. This information then can be integrated into Environmental Decision Support Systems (EDSS) (Poch et al.,2004), that would enable WWTPs to maintain high performance and low emissions at all times, where faults can be acted upon in a timely manner.

The challenge of fault detection in the nitrification oxidation tank

A part of the degradation processes of macro pollutants takes place in the nitrification oxidation tank. In this tank the carbon is oxidized, and the ammonia is converted into nitrate. The process is guaranteed by the insufflation of air into the tank. The control of the blowers is a priority in order to perform a correct and efficient management of the purifier, obtaining high purifying performance at an adequate energy cost. The control of the oxidation and nitrification process is mainly regulated by setting a static oxygen set point and modulating the air flow necessary to maintain the set point. The main limit of this system is that under conditions of low load treated by the purifier, the minimum air flow delivered by the blowers is greater than that required to maintain the oxygen set point with consequent increase of dissolved oxygen and energy waste. As a solution, a control process is used in these tanks (based on the concentration of ammonia nitrogen present in the oxidation tank) that dynamically calculates the oxygen set point to be kept in the tank, arriving to set the set point to zero when the concentration of ammonia decreases below a predetermined value. Although the management of the purification process based on ammonia measurements has shown a great functionality over the years, an erroneous ammonia measurement can lead to non-compliance with the discharge quality required by law or to a high unjustified energy consumption. Therefore, the focus of the proposed work is to detect these types of faults in the ammonia measurements as early and as precisely as possible.

Faults categorisation

In general, faults can be categorized into three groups: i) individual faults, which are unexpected single data instances with respect to other data points; ii) contextual faults that include the individual instances which are anomalous in a specific context and normal in another context; and iii) collective faults, which are manifested through the occurrence of an irregular collection of instances with respect to other data trends (Chandola et al., 2009). The instances in collective faults are not necessarily irregular themselves but a sequence of them is considered anomalous. For instance, when the data points in a sequence happen in an unexpected order or in an unacceptable combination, it is considered as a collective fault. While, several studies have been conducted in using machine learning techniques to detect the first two types of faults in WWTPs sensors, the third and the most complex one, the collective faults have not received enough attention.

Fault detection methods

Apart from categorization of faults, fault detection *methods* can also be categorized into three main groups: statistical methods, learning models, and time series models, in the order of utilization. The most studied methods to monitor WWTPs sensor data are the statistical methods. These approaches range from a simple data trend checking using Mann-Kendall test, to statistical process control (SPC) methods which track process variables of interest over time using statistical control charts. These charts can be univariate such as Shewhart charts, cumulative sum (CUSUM) charts, and exponentially weighted moving average (EWMA) or multivariate methods based on Principal Component Analysis (PCA) (Garcıa-Alvarez, 2009; Padhee et al., 2012) and Kernel PCA (KPCA) (Cheng et al., 2010; Deng and Tian, 2013).
The approaches in the second category, learning models, consider fault detection as a two-class classification problem. Fuzzy classification (Grieu et al., 2001), Support Vector Machines (Fan et al., 2004), Random Forests (Zhou et al., 2019a; Zhou et al., 2019b) and Neural Networks (Hamed et al., 2004; Grieu et al., 2006; Du et al., 2018) are some of the most studied methods in this category. There are several studies on the comparison of statistical and learning methods on wastewater sensor data (Oliveira-Esquerre et al., 2004; Jin and Englande Jr, 2006; Corominas et al., 2018). Neural networks such as Multi-Layer Perceptron (MLP), Self-Organizing Maps (SOM), Radial Bases Functions (RBF) and Functional-Link Neural network are found as the most successful learning methods in fault detection of WWTPs data (Maier and Dandy, 2000).
Both of the above categories can successfully capture the individual faults and contextual anomalies. However, these methods cannot accurately detect complex temporal patterns in collective faults. Therefore, time series modelling methods like ARIMA (Xiao et al., 2017) and Time Delay Neural Networks (TDNN) (Dellana and West, 2009) were introduced to capture temporal patterns in the WWTPs data. ARIMA is a *univariate* linear method that predicts the next data value using the previous data sequence. Subsequently, a conventional control chart is used to plot the prediction error and decide on the normality of data. On the contrary, TDNN is a *multivariate* Neural Network with short-term memory structure, which receives segmented windows of data in time and models non-linear time dependencies of signal (Waibel, 1989). A comparison between linear ARIMA and TDNN is presented in (Dellana and West, 2009) where a clear advantage of TDNN over ARIMA emerges, using eight artificial datasets. However, a shortcoming of TDNN is its dependency on the size of the window to segment the data. The larger the window size is, the higher the dimensions of the network and its parameters become. On the other hand, small window size might not cover all the important information describing system dynamics.

The proposed approach

Recently, deep Recurrent Neural Networks (RNN) such as Long-Short Term Memory networks (LSTM) have shown breakthrough results over state-of-the-art machine learning methods in many applications with non-linear temporal data including robotics, high-energy physics and computational geometry (Goodfellow et al., 2016). These methods can successfully engineer appropriate long-term temporal dependencies and variable length features, significantly lessening the need to pre-process data with respect to traditional machine learning methods or statistical approaches. It is the ability to capture the long-term dependencies that make LSTM networks particularity fitting for the problem at hand.
Although, there is an enormous scope for the possible applications of deep neural networks in management of WWTPs, very few studies (Zhang et al., 2018, 2017) are devoted to this topic

and none have addressed fault detection problem, despite the potential of these methods as highlighted by Sun et al. in their recent review (Sun et al., 2019). This is surprising, considering WWTP operators have vast streams of data at their hands (Corominas et al., 2018), while deep neural networks typically provide highest performance with vast amounts of data. As such, potentially valuable information remains locked in databases, rightfully described as *data graveyards* (Corominas et al., 2018), unexploited and unable to be processed in timely fashion (Yoo et al., 2008).

Main contribution

This work is the first to evaluate a fully automatic fault detection method using a LSTM network, which learns the relevant features in WWTPs sensor data without manual intervention. More specifically, a stacked LSTM network is used to detect collective faults in wastewater sensor data at runtime. While, there have been other work in fault detection methods, such as using Multiparametric Programming (Che et al. 2018), fuzzy neural networks (Honggui et al. 2014), and PCA (Sanchez-Fernández et al., 2015), (Chen et al., 2016), (Carlsson et al. 2016) all these works rely on manual selection of the relevant input features for the corresponding algorithms, typically carried out by the domain expert. This contrasts with the proposed method whereby the LSTM network automatically learns relevant features, consequently reducing domain expert's time and providing superior fault performance detection. The performance of the proposed approach is evaluated on a real-world WWTP dataset gathered in Valdobbiadene wastewater treatment plant in Northern Italy. The dataset contains sensor data spanning a year, where 12 sensors (including chemical and operational sensors) are continuously sampled every minute. Analysis of the resulting dataset of over 5.1 million data samples has shown that stacked LSTM network outperforms all other methods in almost every measure, achieving correct identification of faults (recall) of over 92%. Identifying faults in a timely manner and with high precision will enable increased efficiency in the management of WWTPs, especially in terms of optimising energy use and increasing treatment effectiveness.

The remainder of this paper is organized as follows: the proposed architecture and the LSTM unit are described in following section. Next, the experimental results are presented, while the main conclusions are described in the final section.

Methods

The main objective of the proposed method is to detect collective faults in the WWTP sensor data, considering multivariate, non-linear and temporal behaviour of this data. LSTM based methods have shown breakthrough results in dealing with temporal data such as audio, video, and general time series data. These neural networks can model both long term and short-term correlations in a multivariate data sequence. This section briefly outlines the structure of Long Short-Term Memory nodes along with the architecture of the proposed neural network.

LSTM

Hochreiter et al. firstly introduced Long-Short Term Memory in 1997 as a powerful Recurrent Neural Network for time series prediction (Hochreiter and Schmidhuber, 1997). Basically, a Recurrent Neural Network extracts historical context of the input using a memory cell. The general formulation of an RNN with x_t and h_t as input at time t and hidden state or memory at time t respectively is presented in Equation 1.

$$h_t = \sigma(W^h h_{t-1} + W^x x_t + b) \tag{1}$$

where W^h, W^x, and b are the weights of the hidden state, weights of the input and the bias respectively in which all of them are learned through backpropagation through time. It seems like this approach is good enough for learning long-term sequences as well but Hochreiter et al. (Hochreiter and Schmidhuber, 1997) proved it wrong theoretically and practically due to its exponentially decaying error. Consequently, they offered a solution by adding internal contextual state cells which are able to learn when and what to memorize or forget. To do so, instead of one cell state, they use two cell states, a memory cell C and a hidden cell H. Furthermore, three gates are introduced; I, to process the input and select the addition to the cell state, F to remove unwanted information from cell state, and O to extract the output from what stored in cell state. The LSTM formulation given X as input is provided in Equation 2.

$$\begin{aligned}
I &= \sigma(x_t U^i + s_{t-1} W^i) \\
F &= \sigma(x_t U^f + s_{t-1} W^f) \\
O &= \sigma(x_t U^o + s_{t-1} W^o) \\
G &= \tanh(x_t U^g + s_{t-1} W^g) \\
c_t &= c_{t-1} \circ F + G \circ I \\
s_t &= \tanh(c_t) \circ O \\
y &= softmax(V s_t)
\end{aligned} \tag{2}$$

where W and U are the weights and the biases that should be learned and $\circ$ implies the elementwise multiplication. The overall schema of an RNN unit is compared to LSTM in Figure 1.

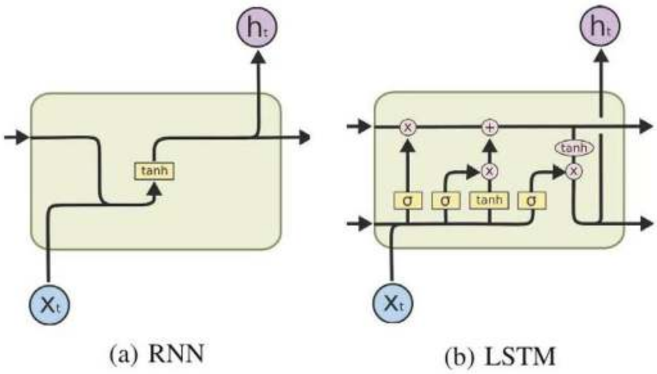


Fig. 1 The general schema of an RNN unit versus a LSTM one (adapted from (Olah, 2015)).

Overall framework

The overall view of the proposed system architecture is presented in Figure 2. The data is gathered from the sensors in the corresponding WWTP to be processed further. Several challenges have been encountered during processing of the data, which are outlined in the subsequent section, followed by a detailed description of the neural network architecture.

Challenges in data processing

Sensor data typically have several challenges that must be addressed before using them in a learning system. The first challenge is the existence of missing values in the data. Poor connection, sensor failures, fading signal strength, are some of the causes. There are number of techniques in the literature of time series data to deal with missing values such as simply ignoring the whole data point with a missing value, filling it with statistically related data or using more complicated methods to estimate the missing value. Since the ongoing research is focused on real-time fault detection, this work follows a less computationally complex approach in which the features with more than 90% of missing values are ignored, while other missing values are filled with the last known value.

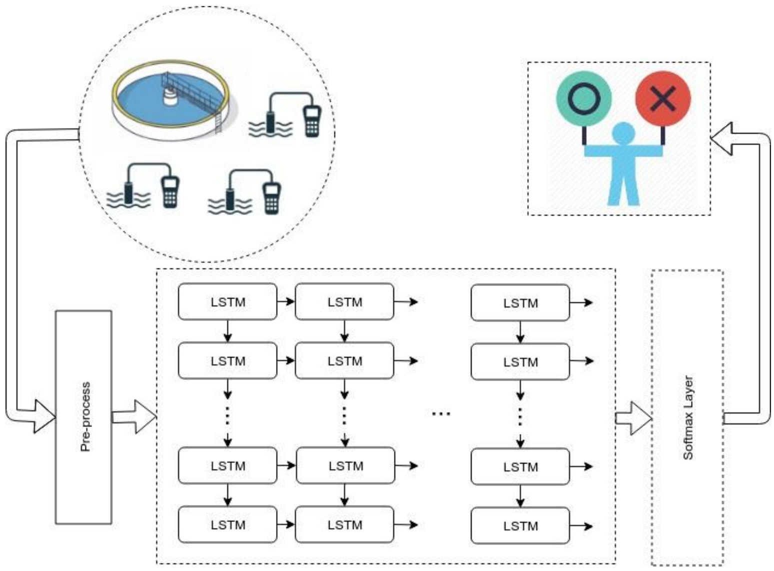


Fig. 2 The overall view of the architecture and the proposed method. The data is gathered from the WWTP sensors and is pre-processed. The data from each sensor is considered a feature in the dataset and the value in each time step is a sample record. These are fed to a multi-layer LSTM network to extract the important features. Finally, the classification layer is used to categorize the data, either faulty or normal.

The other challenge addressed by this work is finding a suitable size of windows used as samples. Sensor data is a continuous time series where the data at each time step is related to its previous values in time. This characteristic of the time series data leads the solution into a recursive approach where a window of data is processed to understand each time step. The window size can greatly influence the performance of the algorithm and therefore should be chosen carefully. A small window can miss the longer relations and large windows can dampen the effect of the short-term relations. This work addresses this problem using LSTM units which receive a relatively large window of data and automatically learn the effective windows of the problem at hand using training data. As mentioned earlier, LSTM units leverage their input and forget gates to control when and what to learn and forget. Therefore, in case of a large window, the unit learns when to replace the old and useless information with the new ones.

Neural Network architecture

As shown in Figure 2, the proposed method consists of stacked LSTM layers for feature extraction and a Softmax layer for classification. The increase in the depth of a neural network results in more abstract features and commonly is attributed as the reason of success in deep learning methods (Hermans and Schrauwen, 2013). This will allow the network to process the data in different time scales.

Considering the output of the pre-processing step in time t as $X = \{X^1, X^2, \dots, X^t\}$ where each element $X^t \in \mathrm{R}^d$ is a d dimensional vector as $X^t = \{x_1^t, x_2^t, \dots, x_d^t\}$ which contains the values from different sensors at time t. The input layer has one unit for each dimension which is fed to the stacked layers of LSTM. In each layer, the unrolled LSTM blocks through time are shown in Figure 2. Each LSTM block receives the vector X^t and process it with several fully connected hidden units inside it. Note that each LSTM layer is succeeded by batch normalization, ReLU activation and Dropout layers.

The data flows in the LSTM layers through time, and the output is a set of carefully extracted features which is given to a softmax classification layer. The output layer has one unit which classifies whether the data sample is faulty or not.

Results and discussion

This section outlines the evaluation of data and its characteristics. Three different models are applied to the dataset including the proposed method. The models' parameters and comparative results are also presented.

Data and labelling

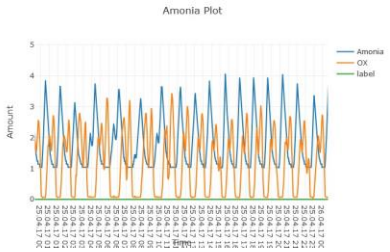
Valdobbiadene is a 10.000 Population Equivalent (PE) sized WWTP located in Treviso province, Italy. Being in the region where Prosecco wine is produced, there is a significant increase of organic mass during the harvest period (late August to early October) reaching 13.000 PE. As such, the aim was to capture not only daily and seasonal variations (typical of WWTP operation) but also other variations that cause significant shifts in plant's load. Consequently, the dataset includes also these load shifts that allowed us to investigate whether the proposed method can capture atypical variations. In this process, data from 12 different sensors (both chemical and operational sensors), including ammonia, are collected from 20/1/2017 to 20/12/2017 at 1-minute intervals. In total there are 438.181 values for each sensor, resulting in over 5.1 million data points (see Table 1).

Table 1 Summary of dataset

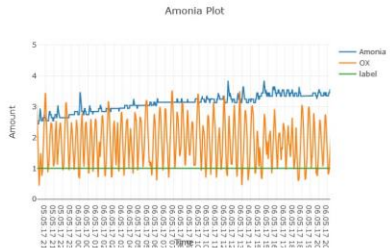
	Instances	Sensor Data	Percent

Normal	376.190	4.514.280	88.5 %
Faults	48.816	585.792	11.5 %
Total	425.006	5.100.072	100 %

The data is labelled by an expert to classify normal and faulty data points. The classification rules are as follows: with the increase in the level of ammonia, the oxygen is released; consequently, the ammonia level decreases, and the oxygen flow is stopped. This cycle is repeated through time. The fault occurs when the ammonia level does not decrease although oxygen is released. An example of normal and faulty behaviour of the data is shown in Figure 3a and 3b respectively where the level of ammonia and oxygen are shown.



(a) Normal behaviour



(b) Faulty behaviour

Fig. 3 A sample of faulty and normal data

Table 2 Description of variables and Spearman correlation with the label (normal or faulty)

Variable	Description	Correlation
AOS	blower frequency	0.24
AUS	blower operational mode	-0.52
DOPLC	PLC operating parameters	-0.13
DOS	blower gear status	0.05
FRS	operating frequency of the blower	0.17
MAS	blower run signal	0.19
NH4	ammonia measurement	0.03
NO3	nitrates measurement	0.24
OX	oxygen measurement	0.25
SP	oxygen set-point	-0.1
Temp	tank temperature	0.15

Description of all sensors (chemical as well as operational) is presented in Table 2 along with the Spearman correlation of each sensor data with the labels (normal or fault). Regardless of the sign, a correlation value shows the strength of association between variables in question. While 'AUS' shows a moderate relation to the label, the other features show insignificant relation with the label and are not individually discriminative enough. Therefore, a multivariate detection algorithm is a necessity to detect these faults which would exclude most traditional univariate statistical methods.

To help the analysis of ammonia better, several statistical measures are extracted from this feature such as Mean, Maximum, Minimum, Variance and Standard Deviation which increase total number of features to 16. The data is segmented to a maximum window size to create the sequences for the LSTM neural network. LSTM network would learn the proper amount of information to learn from this window. The larger the window size is, the higher the dimensions of the network and its parameters would become. On the other hand, small windows might not cover all the important information of system dynamics. Therefore, the size of the window is considered as a hyperparameter for the model and a Grid Search is applied to find the optimal value. The best value is found to be 60 minutes. The samples with at least 10 minutes of faults are labelled as faults and the rest of the data are labelled as normal. 70% of data points are considered as the training set and rest of data points are held for the test set. The statistics of the dataset is summarized in Table 1.

Experiments and evaluation

Four sets of experiments are reported in this section, comparing traditional methods with the proposed method. First, a basic statistical analysis is done on the data. Next, ARIMA, a well-known time series model is applied on the dataset. Then, a learning model using PCA and SVM is also evaluated. The results of the proposed LSTM-based method are presented in the last section. All the settings and parameters are provided in each section. The experiments are implemented in Python programming language using Keras (Chollet et al., 2015) and TensorFlow (Abadi et al., 2015), two open-source neural network libraries designed to build models based on deep neural networks. Keras offers a high-level set of abstractions that make it easier to develop deep learning models and interfaces with TensorFlow as a backend to implement and execute the models.

Variance

Since the faults occur in direct relation to the ammonia level, it is only logical to first analyse this type of sensor data statistically. As the type of fault is known to be collective, the properties of its distribution (mean and variance) change in case of faults. Analysing the mean of the data from ammonia sensors shows that the mean of the data in both normal and fault events are the same. On the other hand, the variance has an apparent difference in these two classes of data. To analyse the variance of data, the segmented 60 minutes of windows are used to calculate the variances and a threshold is set to categorise the window as normal or fault. The threshold is considered as a hyperparameter and it is set based on the training data using a Grid Search. The optimal value is found to be 0.01. The results of this method are shown in Table 4 where it is compared to the other methods.

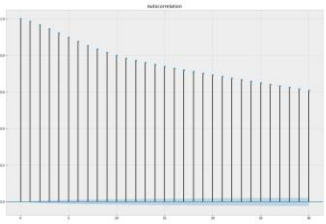
ARIMA

ARIMA, short for AutoRegressive Integrated Moving Average, is a statistical univariate model that learns the normal sequence of a time series to predict its next value in time. This algorithm is widely used as a time series forecasting method (Boyd et al., 2019; Zhang et al., 2019) and a general anomaly detection algorithm for time series data. The ability to detect collective faults on sensor data (Tron et al., 2018; Yaacob et al., 2010; Pena et al., 2013) is tested.

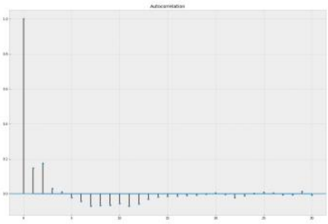
ARIMA is a general form of Moving Average which is applicable only on the stationary sequences. Time series data are stationary if its statistical properties such as mean and variance remain steady over time. ARIMA relies on the idea that a non-stationary data can become stationary by differencing. Particularly, ARIMA assumes that each data point in time series can be derived using a polynomial combination of a number *p*, of its past values which are differenced *d* times plus a number *q* of error variables and a constant *c*, as in Equation 3.

$$Y_t = \varphi_1 y_{dt-1} + \ldots + \varphi_p y_{dt-p} + \theta_1 e_{t-1} + \ldots + \theta_q e_{t-q} + c \tag{3}$$

Therefore, this algorithm can be summarised as *ARIMA(p,d,q)* with three parameters: the AutoRegressive parameter (*p*), the number of differencing steps (*d*), and the moving average parameter (*q*). The algorithm should be trained on the data to learn the coefficients, φ and θ.
Since ARIMA is univariate, the data from ammonia sensor which includes both faults and normal values is set as its input. Next, the predicted value is compared with the previously seen value and in case of meaningful difference, the occurrence of anomaly is reported.
To set these parameters, the Auto Correlation Function (ACF) of the data and its first difference are plotted in Figure 4a and 4b. The plots show a strong correlation between the time series data points and no correlation in the differenced ones. Therefore, the parameter d is set to 1.



(a) The Auto Correlation Function (ACF) of the data



(b) The Auto Correlation Function (ACF) of the data after differencing

Figure 4: The Auto Correlation Function (ACF) of the data and its first difference to set the parameters of ARIMA

For other parameters, p and q, a Grid Search is used to estimate their best values among (0,10). This method searches thorough all possible combinations of p and q in order to obtain minimum Akaike for Information Criterion (AIC). The best parameters are derived as ARIMA (4,1,4) and the model is trained on normal data to set the coefficient for predicting future values. In other words, to predict the next value in the sequence of data, the data from 4 previous steps are integrated once and multiplied by the learned coefficients in addition to 4 error terms with their learned coefficients are all summed up.
Next the ARIMA model is tested on the test data which contains both normal data and faults and the overall Root-Mean-Square Error between the predictions and the real data is 0.07. This result is very good in terms of prediction, but it does not help on detecting faults. The Root-Mean-Square Error is even lower in case of faulty data and the prediction is too exact. Consequently, it is not possible to detect the collective fault behaviour with the ARIMA model in the test data. The main reason is that ARIMA considers only a short-term memory of the data and it does not learn the longer patterns which is a significant factor in detecting collective faults.

PCA and SVM

The fault detection problem can be interpreted as a binary classification of the normal data and the faults. Support vector machines (SVM) are powerful binary classifiers which can be adopted as time series classification method when combined with a feature extraction approach (George, 2012). SVM classifiers simultaneously maximize the performance of the machine, while minimizing the complexity of the model. A variant of this method, support vector regressor, is successfully applied to forecast wastewater quality indicators (Granata et al., 2017). Also, SVM and ARIMA are compared in predicting influent flow rate of a sewage treatment plant in which SVM showed lower error rates (Ansari et al., 2018).
As previously mentioned, the data samples include a window of 60 minutes with 16 features for each minute and consequently the training vectors have more than a thousand feature each. To reduce the feature space PCA (Bo and Wu, 2009; Smith, 2002) method is applied on the data and the data is mapped to lower dimensions in regard to its principal components with the maximum variances. Using PCA improves accuracy while reducing the complexity of SVM model. Furthermore, the unbalanced nature of the data is addressed through the use of weighted SVM.
To evaluate the performance, three measures are calculated for each class: precision, recall and F1 score. These measures are defined as in Equation 4.

$$Precision = \frac{True\ Positive}{True\ Positive + False\ Positive}$$

$$Recall = \frac{True\ Positive}{True\ Positive + False\ Negative}$$

$$F_1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \qquad (4)$$

Since the data is highly unbalanced with 11% faulty data and 89% normal, the learning algorithm is penalized to increase the cost of mistakes in the minority class (fault detection). The final results are presented in next section along with the proposed method in Table 4.

LSTM

As a last step, the proposed LSTM network is trained and tested on pre-processed data. As explained in previous section, the proposed method has several hyperparameters, which are chosen according to the resulted prediction error on the validation set. Random search is used to find the best value for hyperparameters to achieve the lowest prediction error among the following ranges: The number of hidden layers, $h \in \{1,2,3,4,5,6\}$, number of LSTM units in each layer, $u \in \{20,40,60,80,100,120\}$ and the dropout factor, $d \in \{0.2,0.4,0.6,0.8\}$. The best combination is found as 4 layers, 60 units and 0.2 of dropout. Also, rectified linear unit (ReLU) is used as the nonlinear activation function. At each time step, several samples b, are grouped as a batch and fed to the network. Using batch training improves both the learning accuracy and speed. A summary of the network architecture and the number of its learning parameters are presented in Table 3. For each layer, the size of the output matrix is shown as matrix shape where b represents the batch size. The input layer receives b samples of shape 60×16 and passes it to the next LSTM layer with 60 hidden units and 60 time steps.
To the train the network Adam stochastic optimiser (Kingma and Ba, 2014) is used. The batch size is set to 128 examples and the network is trained for 20 epochs using Back Propagation

Through Time with early stopping on the training set. The trained model is applied on the test data and Table 4 illustrates the results.

Table 3 The number of learning parameters of the proposed network in each layer and the total (*b* represents the batch size)

Layer	Output shape	# parameters
Input	(b,60,16)	0
LSTM	(b,60,60)	18240
LSTM	(b,60,60)	29040
LSTM	(b,60,60)	29040
LSTM	(b,60,60)	29040
Softmax	(b,2)	122
Total		105,482

Table 4 Results comparing the proposed method (LSTM) with statistical analysis (Variance) and traditional machine learning methods (PCA - SVM); Highlighted is the best performance of each method on accuracy, F1-score, precision and recall.

Method		VARIANCE	PCA - SVM	LSTM
Accuracy		0.9325	0.9300	**0.9652**
F1-Score	Average	0.8575	0.8667	**0.9267**
	Fault	0.7542	0.7748	**0.8736**
Precision	Average	0.8390	0.8247	**0.9038**
	Fault	0.7067	0.6586	**0.8167**
Recall	Average	0.8796	0.8667	**0.9267**
	Fault	0.8086	**0.9409**	0.9391

Discussion

High detection performance of the tested models, shown in Table 4, highlights the power of machine learning methods in automatic fault detection of real world WWTP data. Since the data is highly unbalanced, accuracy is not the most appropriate measure. Instead, precision, as the classifier's exactness, recall, as the classifier's completeness, and F1-score, as the balance between precision and recall, are considered more accountable. Furthermore, the objective of this work is to minimise missed faults (False Negatives) at the expense of slight increase in false alarms (False Positives). Therefore, the measures on each class are presented separately, highlighting results pertaining to fault detection.

The results show that LSTM network proposed provides superior performance with respect to the other methods considered in this work. This is because LSTM has a high capacity to model complex dependencies between temporal data. Other methods are not well equipped to handle multi-variate time series data and model effectively their dependencies. This ability plays a significant role to detect cumulative faults which have a different pattern in comparison to the typical operational patterns. Furthermore, LSTM is relatively robust to noise and other outliers, which is very common in real-life time series data.

There is a continuous push to improve purification performance of WWTPs while at the same time decreasing energy consumption. This has resulted in increased automation of operation of these plants and, consequently, an increase in the number of measurement sensors. These sensors are being increasingly used, not only for the environmental monitoring, but they are also becoming an important tool in the management of the plants. As such detection of sensors faults is essential in ensuring correct operation of the plant. Furthermore, sensor failure is difficult to be manually detect by the human operator, especially when dealing with large plants with multitude of sensors or small unstaffed plants. While the current systems are very efficient, there is a clear need to develop methods that can reliably detect sensor faults and provide ample time to the plant operators, such that environmental damage is limited when faults occur. A system such as the work presented in this paper, is the first step towards implementing a fully automated fault detection system that can address the issues arising from automatic management of WWTPs.

Conclusions

WWTPs are key infrastructure for the protection of environment. However, being a major energy consumer, it is particularly important to ensure that these plants are operated in a manner that optimises treatment efficiency and energy consumption. One important aspect is detection and management of faults in timely manner. Results presented in this paper have shown that there is a vast potential in using Deep Neural Networks in managing WWTP faults and this work is only the first step in this direction. Not only the proposed method outperformed traditional methods, but the performance achieved in fault detection (recall) of over 92% will enable a new class of WWTP monitoring and management that require very little human supervision. In addition, these methods allow integration with Environmental Decision Support Systems that enable WWTPs to maintain high performance and low emissions, even in response to unexpected events, where faults can be acted upon in a timely manner with minimal environmental impact. It is expected that the work will further encourage the use of Deep Neural Networks, not only in WWTP management, but also in general field of environmental protection.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Man´e, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Vi´egas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., Zheng, X., 2015. TensorFlow: Large-scale machine learning on heterogeneous systems. http://tensorflow.org/.

Ansari, M., Othman, F., Abunama, T. and El-Shafie, A., 2018. Analysing the accuracy of machine learning techniques to develop an integrated influent time series model: case study of a sewage treatment plant, Malaysia. Environmental Science and Pollution Research, 25(12), pp.12139-12149.

Bo, C., Wu, M., 2009. Research of intrusion detection based on principal components analysis. In: 2009 Second International Conference on Information and Computing Science. pp. 116–119.

Boyd, G., Na, D., Li, Z., Snowling, S., Zhang, Q. and Zhou, P., 2019. Influent Forecasting for Wastewater Treatment Plants in North America. Sustainability, 11(6), p.1764.

Carlsson, B.; Zambrano, J. Fault detection and isolation of sensors in aeration control systems. Water Sci. Technol. 2016, 73, 648–653.

Che Mid, E. and Dua, V., 2018. Fault detection in wastewater treatment systems using multiparametric programming. Processes, 6(11), p.231.

Chen, A.; Zhou, H.; An, Y.; Sun, W., 2016. Pca and pls monitoring approaches for fault detection of wastewater treatment process. In Proceedings of the 2016 IEEE 25th International Symposium on Industrial Electronics (ISIE), Santa Clara, CA, USA, 8–10 June 2016; pp. 1022–1027.

Chandola, V., Banerjee, A., Kumar, V., 2009. Anomaly detection: A survey. *ACM computing surveys* (CSUR) 41 (3), 15.

Cheng, C.-Y., Hsu, C.-C., Chen, M.-C., 2010. Adaptive kernel principal component analysis (kpca) for monitoring small disturbances of nonlinear processes. *Industrial & Engineering Chemistry Research* 49 (5), 2254–2262.

Chollet, F., et al., 2015. Keras. https://github.com/fchollet/keras.

Corominas, L., Garrido-Baserba, M., Villez, K., Olsson, G., Cort´es, U., Poch, M., 2018. Transforming data into knowledge for improved wastewater treatment operation: A critical review of techniques. *Environmental modelling & software* 106, 89–103.

Dellana, S. A., West, D., 2009. Predictive modeling for wastewater applications: Linear and nonlinear approaches. *Environmental Modelling & Software* 24 (1), 96–106.

Deng, X., Tian, X., 2013. Nonlinear process fault pattern recognition using statistics kernel pca similarity factor. Neurocomputing 121, 298–308.

Du, X., Wang, J., Jegatheesan, V., Shi, G., 2018. Dissolved oxygen control in activated sludge process using a neural network-based adaptive pid algorithm. *Applied Sciences* 8 (2), 261.

Fan, X.-w., Du, S.-x., Wu, T.-j., 2004. Rough support vector machine and its application to wastewater treatment processes. *Control and Decision*. 19, 573–576.

Granata, F., Papirio, S., Esposito, G., Gargano, R. and de Marinis, G., 2017. Machine learning algorithms for the forecasting of wastewater quality indicators. Water, 9(2), p.105.

Garcıa-Alvarez, D., 2009. Fault detection using principal component analysis (pca) in a wastewater treatment plant (wwtp). In: Proceedings of the International Students Scientific Conference. pp. 1–10.

George, A., 2012. Anomaly detection based on machine learning: dimensionality reduction using pca and classification using svm. *International Journal of Computer Applications* 47 (21).

Goodfellow, I., Bengio, Y., Courville, A., Bengio, Y., 2016. *Deep learning*. Vol. 1. MIT press Cambridge.

Grieu, S., Thiery, F., Traor´e, A., Nguyen, T. P., Barreau, M., Polit, M., 2006. Ksom and mlp neural networks for on-line estimating the efficiency of an activated sludge process. *Chemical Engineering Journal* 116 (1), 1–11.

Grieu, S., Traor´e, A., Polit, M., 2001. Fault detection in a wastewater treatment plant. In: Emerging Technologies and Factory Automation, 2001. Proceedings. 2001 8th IEEE International Conference on. IEEE, pp. 399–402.

Hamed, M. M., Khalafallah, M. G., Hassanien, E. A., 2004. Prediction of wastewater treatment plant performance using artificial neural networks. *Environmental Modelling & Software* 19 (10), 919–928.

Hermans, M., Schrauwen, B., 2013. Training and analysing deep recurrent neural networks. In: Advances in neural information processing systems. pp. 190–198.

Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. *Neural computation* 9 (8), 1735–1780.

Honggui, H.; Ying, L.; Junfei, Q. A fuzzy neural network approach for online fault detection in waste water treatment process. Comput. Electr. Eng. 2014, 40, 2216–2226

Jin, G., Englande Jr, A., 2006. Prediction of swimmability in a brackish water body. *Management of Environmental Quality: An International Journal* 17 (2), 197–208.

Kingma, D. P., Ba, J., 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

Maier, H. R., Dandy, G. C., 2000. Neural networks for the prediction and forecasting of water resources variables: a review of modelling issues and applications. *Environmental modelling & software* 15 (1), 101–124.

Olah, C., 2015. http://colah.github.io/posts/2015-08-Understanding-LSTMs.

Oliveira-Esquerre, K. P., Seborg, D. E., Bruns, R. E., Mori, M., 2004. Application of steady-state and dynamic modeling for the prediction of the bod of an aerated lagoon at a pulp and paper mill: Part i. linear approaches. *Chemical Engineering Journal* 104 (1-3), 73–81.

Olsson, G., Newell, B., 1999. *Wastewater treatment systems*. IWA publishing.

Padhee, S., Gupta, N., Kaur, G., 2012. Data driven multivariate technique for fault detection of waste water treatment plant. *International Journal of Engineering and Advanced Technology* 1, 45.

Pena, E. H. M., de Assis, M. V. O., Proena, M. L., Nov 2013. Anomaly detection using forecasting methods arima and hwds. In: 2013 32nd International Conference of the Chilean Computer Science Society (SCCC). pp. 63–66.

Poch, M., Comas, J., Rodríguez-Roda, I., Sanchez-Marre, M. and Cortés, U., 2004. Designing and building real environmental decision support systems. Environmental Modelling & Software, 19(9), pp.857-873.

Sanchez-Fernández, A.; Fuente, M.J.; Sainz-Palmero, G.I. Fault detection in wastewater treatment plants using distributed pca methods. 2015 IEEE 20th Conference on Emerging Technologies & Factory Automation (ETFA), Luxembourg, Germany, 8–11 September 2015; pp. 1–7

Shewhart, W. A., 1931. Economic control of quality of manufactured product. ASQ Quality Press.

Smith, L. I., 2002. A tutorial on principal components analysis. Tech. rep., Department of Computer Science, University of Otago, New Zealand.

Sun, A.Y. and Scanlon, B.R., 2019. How can big data and machine learning benefit environment and water management: A survey of methods, applications, and future directions. Environmental Research Letters.

Tron, T., Resheff, Y. S., Bazhmin, M., Weinshall, D., Peled, A., 2018. Arima-based motor anomaly detection in schizophrenia inpatients. In: Biomedical & Health Informatics (BHI), 2018 IEEE EMBS International Conference on. IEEE, pp. 430–433.

Villez, K., Vanrolleghem, P. A., Corominas, L., 2016. Optimal flow sensor placement on wastewater treatment plants. *Water research* 101, 75–83.

Waibel, A., 1989. Modular construction of time-delay neural networks for speech recognition. *Neural computation* 1 (1), 39–46.

Xiao, H., Huang, D., Pan, Y., Liu, Y., Song, K., 2017. Fault diagnosis and prognosis of wastewater processes with incomplete data by the auto-associative neural networks and arma model. *Chemometrics and Intelligent Laboratory Systems* 161, 96–107.

Yaacob, A. H., Tan, I. K. T., Chien, S. F., Tan, H. K., Feb 2010. Arima based network anomaly detection. In: 2010 Second International Conference on Communication Software and Networks. pp. 205–209.

Yoo, C. K., Villez, K., Van Hulle, S. W., Vanrolleghem, P. A., 2008. Enhanced process monitoring for wastewater treatment systems. Environmetrics: *The official journal of the International Environmetrics Society* 19 (6), 602–617.

Zhang, D., Hølland, E. S., Lindholm, G., Ratnaweera, H., 2017. Hydraulic modeling and deep learning based flow forecasting for optimizing inter catchment wastewater transfer. *Journal of Hydrology*, 567, 792-802.

Zhang, D., Holland, E. S., Lindholm, G., Ratnaweera, H., 2018. Enhancing operation of a sewage pumping station for inter catchment wastewater transfer by using deep learning and hydraulic model. arXiv preprint arXiv:1811.06367.

Zhang, Q., Li, Z., Snowling, S., Siam, A. and El-Dakhakhni, W., 2019. Predictive models for wastewater flow forecasting based on time series analysis and artificial neural network. Water Science and Technology, 80(2), pp.243-253.

Zhou, P., Li, Z., Snowling, S., Baetz, B.W., Na, D. and Boyd, G., 2019a. A random forest model for inflow prediction at wastewater treatment plants. Stochastic Environmental Research and Risk Assessment, 33(10), pp.1781-1792.

Zhou, P., Li, Z., Snowling, S., Goel, R. and Zhang, Q., 2019b. Short-Term Wastewater Influent Prediction Based on Random Forests and Multi-Layer Perceptron. JOURNAL OF ENVIRONMENTAL INFORMATICS LETTERS, 1(2), pp.87-93.