DOI: 10.48011/asba.v2i1.1436

# Towards the use of LSTM-based Neural Network for Industrial Alarm Systems

**Márcio J. da Silva and Carlos E. Pereira**

*Department of Automation and Energy*
*Federal University of Rio Grande do Sul, Porto Alegre, RS, Brazil,*
*(e-mail: marcio.silva@ufrgs.br, cpereira@ece.ufrgs.br).*

**Abstract:**
With the coming of the complex industrial plants, the majority of control rooms sharing a common scenario involve operators faced amount of alarms during an overload. Besides, were often unable to determine which were important in an alarm widespread flooding scenario is an increasingly frequent situation. To solve this issue, proper handling of industrial processes requires designing, implementing and updating *Process Malfunction Prediction* (PMP) systems able to give advice to the operators being a recommendation to make necessary adjustments in operating variables. From the wide range of models, we apply *Recurrent Neural Networks* (RNNs) using *Long Short-Term Memory* (LSTM) units, built was a regressor trained to predict the behavior of failures in an industrial process. The activity values of LSTM units can give recommendations for the monitoring of malfunctions. To this end, data mining and machine learning techniques are used, which allow the implementation of a regression. The distinctive feature of PMP is that dynamically provides information using the data process. Further, proposed approach was evaluated in a simulated industrial process case study scenario. Lastly, the evaluation of the experimental results demonstrate the contribution of this work.

*Keywords:* Regression, Recurrent Neural Network, Long Short-Term Memory, Process Malfunction Prediction, Industrial Alarm Systems

## 1. INTRODUCTION

An alarm should be triggered when a process variable exceeds their corresponding control limits (Izadi et al. (2009), Habibi and Hollifield (2006)). To understand how operators monitoring processes through devices as a *Human-Machine Interface* (HMI) (ANSI/ISA-101.01:2015. (2015), ISO-11064-5:2008. (2015)) might approach such alarms and events, it is important to have some knowledge about how control and monitoring practices in process industries changed after the advent of a *Distributed Control System* (DCS)-based annunciators (Kazemi et al. (2019)). More recently, these alarms are created or changed by configuring a setting in a *Supervisory Control and Data Acquisition* (SCADA). An alarm requires operator action, and the time for human analysis is short in contrast with the amount of generating data. Moreover, the emphasis is on the actions taken in response to normal and upset contexts as shown in Figure 1. Nevertheless, during an abnormal scenario, an operator may be required to perform a series of actions in time (Silva et al. (2019)).

The industrial automation level achieved in the last decades has often been scared against to derive insight from their event generated. In industry, an event consists of any relevant occurrence within the operational scope of the system. Events detail the general plant operation circumstances and generally do not require acknowledgement or intervening actions. Alarms, in turn, are audible or visible means of indicating an equipment malfunction, deviations in the process or abnormal conditions, requiring a response from operators as shown in Figure 1.

According to ANSI/ISA-18.2:2016. (2016), the alarms must be configured to inform only the most necessary events, following a prioritization that can be carried out through rationalization, since alert rates can exceed levels that can be managed by the human being. Regarding Three Mile Island accident in 1979, or the Milford Haven in 1994, or on March 20, 2001, the Petrobras Platform 36 (P36) sink into the Atlantic Ocean (NASA (2008)). The series of alarms and events, which are explained by the poor performance of alarm systems, human factors are not taken into consideration during the design or aid the control operators in addressing the overload situations and further incidents (Wang et al. (2016), Kourti (2002)).

According to the *Engineering Equipment and Materials Users Association* (EEMUA), the purpose of an alarm system is to redirect the operator's attention towards plant conditions requiring timely assessment or actions (EEMUA-191:2007. (2007)). This huge amount of data is used for control and monitoring of the plant (Schneider et al. (2017)). Although many variables are measured in a plant, they are not all independent. A single fault could cause many system signals to exceed their limits and appear as multiple faults, and hence the fault isolation is very difficult. According to (Frank and Blanke (2007)), faults occur in the industrial processes that cause undesired or unacceptable system behavior. For

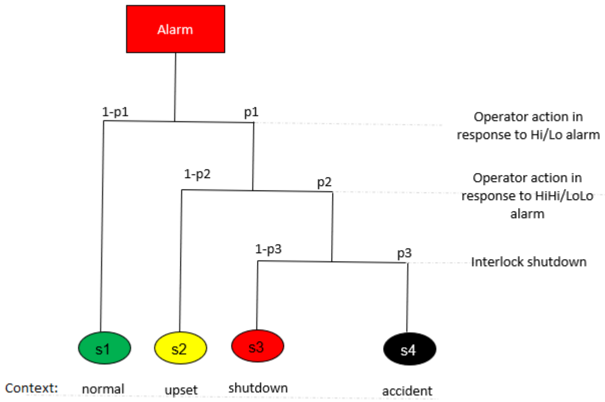


Figure 1. Illustration of the alarm with labels for each context.

instance, in hazardous processes as e.g. chemical plants, the consequences of a fault can be disastrous.

Faults in devices as sensors represent discrepancies between the measured and real values of system variables. Actuators fault, represent discrepancies between input commands for actuators and their real output (Zaytonn and Lafortune (2013)). Consider how the alarm system might be designed to handle events. A fault can generate an alert signal as alarms and hundred discrete data are generated. According to the ISA 18.2 and EEMUA 191, an alarm annunciation rate should be no more than 10 alarms per 10 minutes. By analysing the industrial process, after collecting the historical data, and choose a witch fault detection model is more useful. The model costs and performance values to ensure the best cost-effective relation.

The detection of failures in industrial systems is closely linked to the temporal relations that occur in the process (Silva et al. (2019)). This work proposes a study of the literature concerning the use of AI based on techniques to pattern recognition. In short, Recurrent neural network layer with LSTM units was applied from the wide range of models used for time series analysis after defining a sliding window. The new method is based on the behavior of process variables, using multi-temporal sequences of input signals will later be applied as inputs of LSTM (Yang et al. (2019), Tian et al. (2018)).

The rest of the paper is structured as follows. Section 2 provides a review of the literature and describes the background. Section 3 provides a graphical representation of the event based environment. Section 4 presents the relevant information in order to implement the proposed approach. Section 5 presents the results, while discussing these results and their application to actual setups. Section 6 concludes the paper providing directions for future work.

## 2. BACKGROUND AND RELATED WORK

A paradigm of *Artificial Intelligence* (AI) is the *Artificial Neural Networks* (ANNs) formed by a set of neurons and their interconnections used for regression and classification (Jia et al. (2016)), which are responsible for processing information. Artificial neural networks comprised in the AI paradigm that seeks the solution of problems through the computational simulation of the mechanisms and structures of the human brain. A neural network consists basically in the interconnection of artificial neurons, forming a mesh composed of some or several signal processing units. The ANN structure adopted, implies the learning algorithm directly to be used. It is through the learning algorithm that ANN obtains the knowledge necessary to solve the problem (Haykin (1998)). Consequently, this algorithm ultimately determines the ability to detect faults (Himmelblau (1980), Moseler and Isermann (2000), Blanke et al. (2006), Dubrova (2013), Venkatasubramanian et al. (2003)).

A context knowledge can be further applied in a PMP. The working hypothesis is that the context should improve performance by a perspective from the one presented in (Graves et al. (2005) and Greff et al. (2016)).

RNNs are designed to capture time-dependency in sequential data. These models were popular after the introduction of RNNs with LSTM units, proposed to overcome the difficulties of handling long-term dependency and vanishing gradient (Razvan et al. (2013), Sepp and Jürgen (1997)).

According to (Greff et al. (2015)) the central idea behind the LSTM architecture is a memory cell which can maintain its state over time, and non-linear gating units which regulate the information flow into and out of the cell.

The LSTM is structured and the results are compared even with a distinct activation function where each visible unit has a bias scalar multiplication of the activation function input.

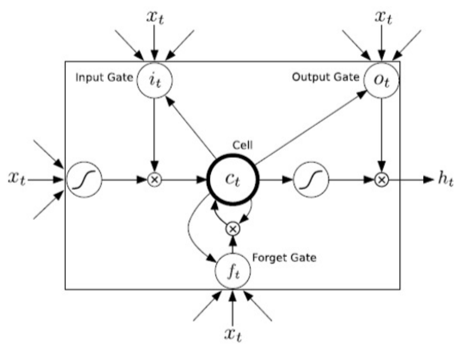


Figure 2. Structure of LSTM cell. Source: Adapted from (Sepp and Jürgen (1997)).

The activation function can be expressed as (Su et al. (2017), Ravanbakhsh et al. (2016), Martens et al. (2013)):

$$Sigmoid = \frac{1}{(1+e^{-x})}, \quad tanh = \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}}$$

LSTM cell as shows in the Figure 2 captures the temporal relationship among time series data. The computation can be defined as follows (Yang et al. (2019)):

$$f_t = Sigmoid(W^f h_{t-1} + b^f) \tag{1}$$
$$i_t = Sigmoid(W^i h_{t-1} + b^i) \tag{2}$$
$$c_t = tanh(W^c h_{t-1} + b^c) \tag{3}$$
$$m_t = \mathrm{f}_t \odot m_{t-1} + i_t \odot c_t \tag{4}$$

$$o_t = Sigmoid(W^o h_{t-1} + b^o) \quad (5)$$

$$h_t = tanh(o_t \bigodot m_t) \quad (6)$$

where:

$W^i, W^f, W^o, W^c$ are weight matrix-vector multiplication by $(h_{t-1})$, and $b^i, b^f, b^o, b^c$ are the bias, and $o_t \bigodot m_t$, $f_t \bigodot m_{t-1} + i_t \bigodot c_t$ are element-wise multiplication.

Regarding the data mining, known as the knowledge discovery task that has many applications (Ge et al. (2017)). A practical implementation of this process runs through several steps until by training data, it is possible to generate a regression. In this work, statistical and machine learning methods are applied to databases generated from a simulated industrial process.

The technological evolution of automation systems has been accompanied by the exponential growth of data of processes. The work in (Yu et al. (2017)), presents a method for detecting abnormal data segments for historical data samples, noting the correlation directions between related process variables. However, the study is finalized without discussing how to implement this method in an industrial process effectively to predict an abnormal situation.

According to (Sarnovsky et al. (2018)), predictive models can offer effective gains when applied to the optimization of productive processes. At this point, it is important to enlighten that masses of data generated are alarm and event logs are a potential source of knowledge that can be better explored.

The learning of the machine, and intelligence where there are different computational methods related to the task of knowledge discovery. One of the known categories of learning supervision and supervised learning, a paradigm.

There are four sections of models in machine learning which are used depending on what input and output data is available. Supervised learning has a network in which the current output is compared to the desired output, from the input patterns, showing the direction of adjustment of the network weights, to guarantee the smallest error between the network output and the desired output. In unsupervised learning, input patterns are presented continuously and the regularity between these data allows for learning. However, the data label is not known in this process. This learning ability of ANNs networks allows us to perceive data patterns without the need to know the equations or models that gave rise to these data. Semi-supervised learning occurs when only a small amount of the inputs has a corresponding output. In the end, reinforced learning occurs when no historical input or output is available, the parameter learning is then based on a trial and error reward system (Russell and Norvig (2003)).

The use of artificial intelligence is applied in the diagnosis of faults given the lack of an effective analytical formulation capable of solving this problem. The detection of failures in industrial systems is closely linked to the temporal relations that occur in the process.

The work in (Varga et al. (2010)) presents a model based on predictive stability analysis to detect a dangerous situation in advance. The methodology involves the anticipated simulation of the effect of adjustments in a chemical process performed by the operator and calculates a time where each action must be performed to avoid a disturbance in the plant. According to the authors, there is a last controllable operation point at each manipulation before the process equipment becomes unstable and is presented with a prediction-based model to detect certain situations in a chemical industrial process.

In (Nashalji et al. (2014)), the authors use a combination of the detection technique of neural networks. The work focuses on the reduction of dimensionality as a way to improve results obtained with the neural classification networks. The authors are using two topologies structures in the neural networks, which are evaluated separately. The first containing classification of the sample in binary form and the second as a diagnostic of failure, describing in which class the sample belongs.

The work in (Wang et al. (2016)) presented an overview of industrial alarm systems and shows the main causes for alarm overloading and the most common way to detect an alarm state among related process variables.

The work in (Wang et al. (2017)) presented a method to reduce the number of nuisance alarms among related process variables using techniques to detect specific types of nuisance alarms and designing delay timers. However, the study ends without discussing machine learning implementations in an effective industrial process to predict an abnormal situation.

In (Yin et al. (2012)), statistical measures with manually set thresholds are used, that make the use on a large scale difficult for an individual analysis is required for each class of failure. The results obtained from the cited work present space for improvements, there is considerable variation in all of them between fault detection depending on the type of fault analyzed.

In (Haitao et al. (2018)), the authors propose a fault diagnosis method based on LSTM evaluated Tennessee Eastman benchmark with a classifier approach and a batch normalization to improve the convergence.

This work presents the results of the implementation of a PMP method for analysis of data generated in a simulated industrial process.

## 3. STUDY ENVIRONMENT AND SITUATIONS

An increase in the sophistication of process control systems has not eliminated abnormal situations. Consequently, faults in a complex environment, like the factory floor, are inevitable. This fault may cause from a reduction in the process performance to injuries in workers. Systems that have an imperative need for disponibility and reliability apply fault tolerance techniques by the need to maintainability, performance in situations where the controlled system can have potentially damaging effects on the environment.

The emergence of new situations over time, which the mechanism has never experienced, and which are related to risk factors with high priority, it is necessary to determine during moments in which proactive actions prevent the occurrence of the situation like an information overload. This is a severe problem in control process rooms and operators of such systems are at risk lacking situation and systems focus on the operation panel causes confusion between operators during a process anomaly.

The historical incidents have become serious accidents as discussed in Section 1. Often either industrial process operators are kept unaware of shutdown due to the failure and operational missteps. In the case where multiple alarms may be triggered at once, the operators need a real-time methodology to determine the priority of each alarm in order to address them effectively and in the proper sequence (NASA (2008)). This scenario causes information overload to the operator and can occur even if there is a traditional alarm system (Silva et al. (2018)).

August 2008, the Bayer CropScience pesticide process plant accident started as shown in Figure 3 in a chemical reaction runaway. Operators scrambled to shutdown emergency vessel. After this, a set off a violent pressure that exploded, killing and injuring people. The alarms were triggered in a few minutes after the first indication of the high-pressure alarm. In the hazardous process and risk to people, public and environment, a controlled shutdown was required in a safety instrumented system (SIS) (ANSI/ISA-18.2:2016. (2016), CSB (2011)).

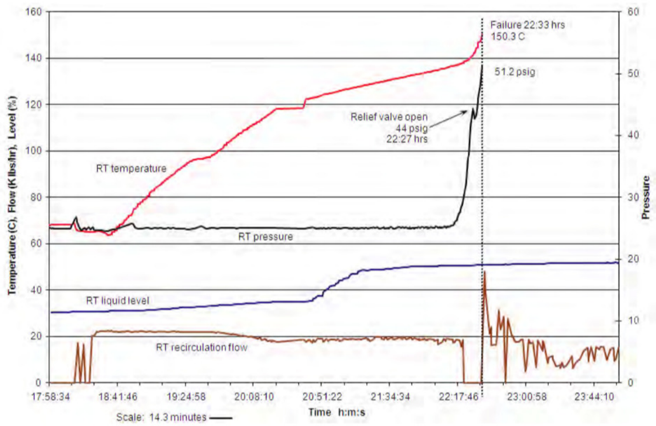


Figure 3. Process variable behavior before the explosion. Source: Adapted from (CSB (2011)).

The mass of data generated in the automation of process variable data from field devices includes another kind of data that also comes from plants in great amount: alarm and event logs. Also closed-related to plant processes, these records are bond to plant operation monitoring, and commonly are brought to the attention of operators. According to (Soares et al. (2016)), in an industrial process monitoring, there is a correlation between variables during an abnormal situation and that can affect other variables simultaneously.

## 4. METHODOLOGY

In this section, the methodology developed and used will be presented. The goal of this process is capturing relevant information in order to make the predicted behavior easier of the machine learning algorithms of faults in the process. The Programming language is Python 3.6 with Keras using Tensor Flow backend (library for Theano and TensorFlow (2020)). The subsections below present a description of the approaches used.

### *4.1 Data Generate*

The study base uses a simulated data process (A). The study base generated has attributes that correspond to the various measures of the process. The process with several responsible for the different stages of production. The input variables have parameters as the sensor value and this directly impacts on the mode of operation of the plant. From the values of the different signals from this process and that the data are generated are used in this study. We stored the output values as a CSV format file. This file has columns that represent the data. Rows represent an observation of all values. Faults can be specified and saved, to generate an annotated database which is used in the mining step proposed in this study.

### *4.2 Data Preprocessing*

For the development presented in this paper a set of open-source tools in Python libraries were used, with its respective display packages, statistics, data manipulation and learning of the machine for the execution of preprocessing tasks, data mining and deep learning.

The second step is where we preprocessing data (B). After importing data it is necessary to define the training set that contains input data to use in RNN.

```
.
.
X_train = [ ]
y_train = [ ]

for i in range(100, 1000):
    X_train.append(Data_train_set_W[i-100: i,0])
    y_train.append(Data_train_set_W[i,0])
.
.
.
```

Listing 1: Specific data structure.

A normalization step is necessary to use in Sigmoid as the activation function in the deep learning process. It is necessary to define the feature range between zero and one because all the new values are in this range. After this, it is necessary to create a new data structure to apply this normalization as a new variable that contains the train data set normalized between zero and one.

The third step is where we give our contribution whit the creation of a specifying data structure and specifying what the RNN can be remembered to predict new values using a sliding window, shown in Listing 1, defined by the right time steps. These time steps mean that at each time $t$ the RNN is going to look at the values before the defined time step and based on the tendencies and correlations during this time step it was to predict the next value in $t+1$. Defining the correct number of time steps is essential to prevent overfitting.

Following, we apply the prepared data to RNN. After the model training, we have the reshaping of the data that adding dimensionality of the structure data build in the last step, called a unit, that represents the number of predictors that we used to predict the next value. At this time, we will be adding more indicators to help to predict the ascendent and descendent tendencies of the variables. To add this new dimension to the matrix, it is necessary to use one more reshape function with 3D tensor, with the shape as the number of observations, the time steps, and the predictors.

### *4.3 Building the RNN*

The RNN (C) was a regressor shown in Listing 2, to predict the features that give us a better data representation of the process behavior using Tensor Flow. To predict a continuous value, we use a regression. After this, we add to the LSTM layers some dropout regularization.

The step (C) is where we give our contribution. We apply RNN to take the predictions and this choice gives us the possibility to explore the regression algorithm. After regression model training, we have the validation phase.

The optimizer Adam was used at this step (C). For Regression, the way to evaluate the model performance is with a metric called *Root Mean Squared Error* (RMSE) (Kofi et al. (2013)). It is calculated as the root of the mean of the squared differences between the predictions and the real values.

```
.
.
.
#initializing the RNN

regressor = Sequencial()

# first layer

regressor.add(LSTM(units=50, return_sequences=True,
input_shape= (x_Data_train.shape[1], 1)))
regressor.add(Dropout(0.2))

# second layer
regressor.add(LSTM(units=50, return_sequences=True))
regressor.add(Dropout(0.2))

# third layer
regressor.add(LSTM(units=50, return_sequences=True))
regressor.add(Dropout(0.2))

# fourth layer
regressor.add(LSTM(units=50))
.
.
```

Listing 2: The RNN implemented as a regressor.

Observing the benefit in predicting up and down tendencies of process variable values using LSTM layers. For finishing this step, it is necessary to connect the network to the train data set and executing the training over the number that we choose on fitting method and defined the number of epochs to converge and the batch size number to define when updating the weights in the artificial neural network.

### *4.4 Make Predictions*

The train and test split (D) is a technique where part of the data set is used for training the models and another part is used to validate these models. Having the validation results, we can work with the parameters (C) to achieve the best performance. At the end of this improvement loop, we go to deploy the phase (D). The deploy phase is out of scope of this work.

The contributions of this work can be shown in the practical analysis and evaluation insights. The tests were performed to evaluate the behavior of the process variables. This work presented the application of machine learning implemented in python, applied in the data analysis of an industrial process.

## 5. RESULTS, DISCUSSION, AND FURTHER INVESTIGATION

In this section, we present the results of the tests. The performed tests aim to identify the behavior of continuous process variables. Data used for these initial tests comes from a database by the simulated industrial environment.

The raw input data that came from multiple sensors in an industrial environment are complex and full of noise. All cited works face this problem, and make an effort to extract useful information from the input data to perform the fault behavior detection with better results.

Our initial tests use the Tennessee Eastman Process (TEP) to simulate an industrial environment. This simulation generates data regarding normal operation and multiple simulated faults and stores it in a Database (A). TEP implementation generates the test dataset used as input to our regression model. We configured the simulation to cover the operation normal state and all 20 faults states, and stored the output values as a CSV format file. This file has columns that represent the processes features, like internal reactor pressure, temperature or the aperture that controls the mixing of the elements. Rows represent an observation of all 53 Tennessee Eastman Process features. Faults can be specified and saved, to generate an annotated database which is used in the mining step proposed in this study.

The study base generated has 40 attributes that correspond to the various measures of the process as shown in Figure 4. The process simulates a chemical process with several responsible for the different stages of production. This process has four reagents, two products, and one by-product (Downs and Vogel (1993)). The input variables have parameters as the flow of reagents and this directly impacts on the mode of operation of the plant.

By understanding the situations during production can significantly improve the operator's ability to manage. The core of abnormal situations are approaches that address root causes. Several advantages can be identified from applying the PMP. The use of machine learning algorithms in applications related to predicting process variable in failure behavior. We successfully apply the methodology

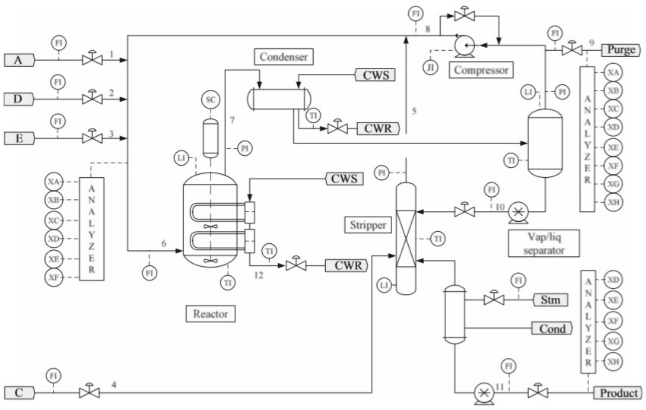


Figure 4. The Tennessee Eastman Process flow sheet. Source: Adapted from (Downs and Vogel (1993)).

to predict failure behavior on the related process. We demonstrate that it will enable novel intelligent methods to improve process operation in the future, for example, by forwarding fault propagation estimating the effect of a control logic failure propagation where multiple alarms may be triggered.

In this section, we present the results of the tests. The performed tests aim to identify the behavior of continuous process variables. Data used for these tests comes from a sequence of process variables. Moreover, generated data, results from the simulated industrial process, using MATLAB® / Simulink® software. At the end, performed experiments using a database by the industrial environment.

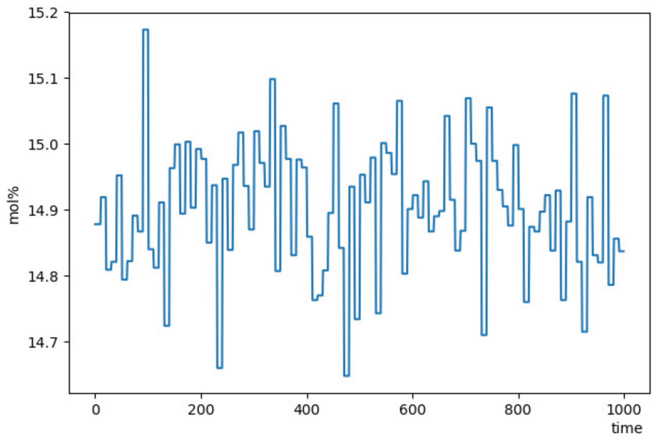


Figure 5. The XMEAS 24 in normal behavior.

Figure 5 shows the normal behavior of the element inlet flow of the reactor, where it is possible to note that the value is stable around the value of 14.9 % of total mass entering the reactor.

A predictive analysis of process variable XMEAS 24 (Downs and Vogel (1993)) in normal behavior is presented in Figure 6. An industrial process accumulates data that will support valuable conclusions regarding current production that can be useful for predicting situations.

Figure 7 shows the failure behavior, there is an increase the concentration of this element at the entrance of the reactor, reaching the value of 16 %. This is an important

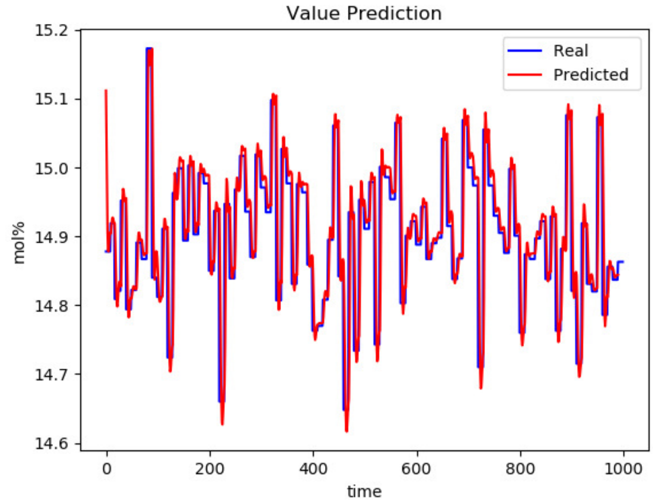


Figure 6. A predictive analysis of process variable XMEAS 24 in normal behavior.

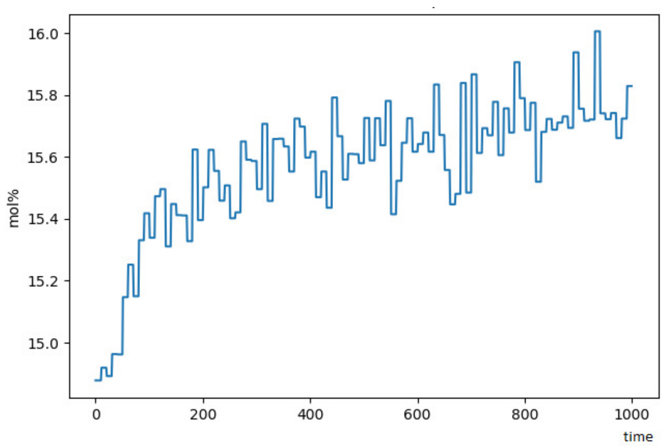


Figure 7. The XMEAS 24 in failure behavior.

functional requirement for alarm system it must detect and warn the operator of abnormal operating condition that require attention. In this situation, we must rely on either an operator quickly identify the situation and initiating an emergency state.

The conducted experiments aimed to assess the behavior of the continuous process variables as shown in Figure 8 where it is possible to note that the value is predicted around the value of XMEAS 24 in failure behavior presented in Figure 7.

Our solution uses the Tennessee Eastman Process to simulate an industrial environment. This simulation generates data regarding plant operation. According Table 6 in (Downs and Vogel (1993)), the reactor has a normal pressure operating limit of 2895 kPa. The conducted experiments aimed to assess the behavior of the process variable XMEAS 24 in failure behavior shown in Figure 8 during the increase the concentration of this element at the entrance of the reactor make a step change so that the reactor operating pressure changes as shown in Figure 9.

The use of AI should be applied in the diagnosis of faults given the lack of an effective analytical formulation capable

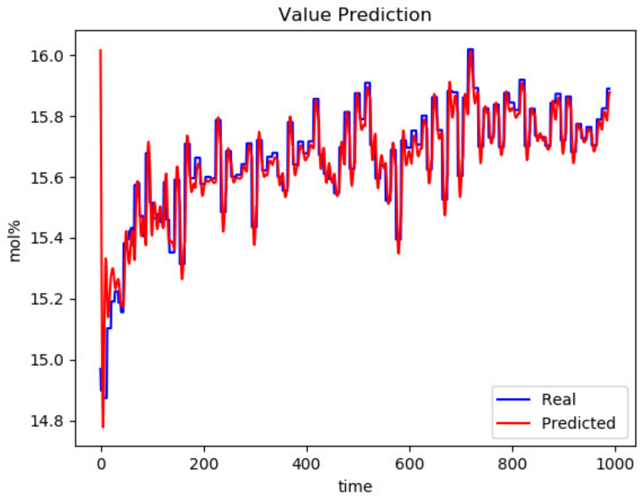


Figure 8. A predictive analysis of process variable XMEAS 24 in failure behavior.

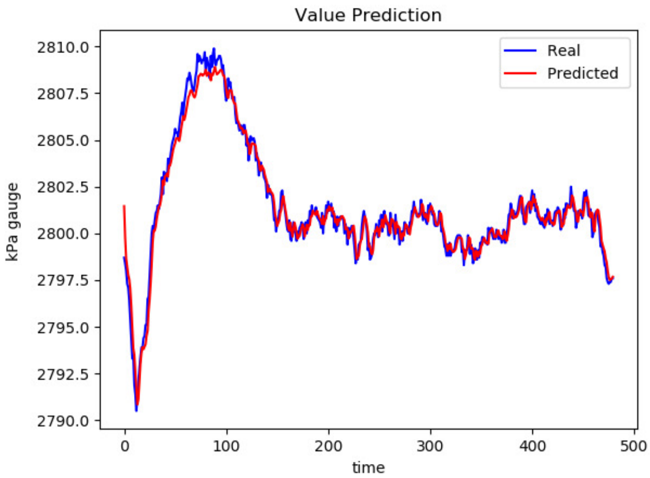


Figure 9. A predictive analysis of reactor pressure.

of solving this problem. This work proposes a study of the literature for the detection of fault behaviour.

## 6. CONCLUSION

Most of the malfunctions that lead to failure and consequently a stop are process driven and most of them can be predicted and even controlled. This paper was engaged in the prediction of process variables in an industrial process. Corroborating with this, the methods were based on learning the supervised machine. For the training of regressor, data were used for training and validation extracted from the database.

The results obtained through the computational analysis showed that the regression method proposed in this work can efficiently identify the failure's behavior that has occurred in this process. This is a critical issue, especially, because or the faults, requiring action or analysis at a specific time. In this way, it is possible to evaluate the behavior of events and to predict the situation of the plant, being also possible to generate recommendations.

## ACKNOWLEDGMENTS

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001

## REFERENCES

ANSI/ISA-101.01:2015. (2015). *Human Machine Interface for Process Automation Systems.* Available at: `http://www.isa.org`.

ANSI/ISA-18.2:2016. (2016). *Management of Alarm Systems for the Process Industries. The International Society of Automation.* Available at: `http://www.isa.org`.

Blanke, M., Kinnaert, M., Lunze, J., Staroswiecki, M., and Schröder, J. (2006). *Diagnosis and fault-tolerant control*, volume 691. Springer.

CSB (2011). Investigation report: Pesticide chemical runaway reaction pressure vessel explosion. report no. 2008-8-i-vw. Available at `http://www.csb.gov`.

Downs, J.J. and Vogel, E.F. (1993). A plant-wide industrial process control problem. *Computers & chemical engineering*, 17(3), 245–255.

Dubrova, E. (2013). *Fault-tolerant design.* Springer.

EEMUA-191:2007. (2007). *Alarm Systems: A Guide to Design, Management and Procurement.* Available at: `http://www.eemua.org/Products/Publications/Print/EEMUA-Publication-191.aspx`.

Frank, P. and Blanke, M. (2007). Fault diagnosis and fault-tolerant control. *Control Systems, Robotics and Automation*, 16.

Ge, Z., Song, Z., Ding, S.X., and Huang, B. (2017). Data mining and analytics in the process industry: The role of machine learning. *IEEE Access*, 5, 20590—20616. doi:10.1109/ACCESS.2018.2794765.

Graves, A., Jürgen, S., and Souza, M.B. (2005). Framewise phoneme classification with bidirectional lstm and other neural network architectures. *Neural Networks*, 18, 602–610. doi:10.1609/aimag.v17i3.1230.

Greff, K., Srivastava, R.K., Koutník, J., Steunebrink, B.R., and Schmidhuber, J. (2016). Lstm: A search space odyssey. *Transactions on Neural Network and Learning Systems.*

Greff, K., Srivastava, S., Koutnik, J., Steunebrink, B.R., and Schmidhuber, L. (2015). Lstm: A search space odyssey. *Neural Computation.*

Habibi, E. and Hollifield, B. (2006). Alarm systems greatly affect offshore facilities amid high oil process. *World Oil Magazine, Houston, v.*, 227, 1–4.

Haitao, Z., Shaoyuan, S., and Bo, J. (2018). Sequential fault diagnosis based on lstm neural network. *IEEE Access: Special Section on Sequential Data Modeling and its Emerging Application*, 6. doi:10.1109/ACCESS.2018.2794765.

Haykin, S. (1998). *Neural Networks: A Comprehensive Foundation.* Prentice Hall, 2 edition.

Himmelblau, D.M. (1980). Fault detection and diagnosis in chemical and petrochemical processes. *The Chemical Engineering Journal,*, 20, 79–85.

ISO-11064-5:2008. (2015). *Ergonomic design of control center - Part 5: Displays and Control.* Available at: `http://www.iso.org`.

Izadi, I., Shah, S.L., and Shook, D.S. (2009). An introduction to alarm analysis and design. *Proceedings of the*

*7th IFAC Simposium on Fault Detection, Supervision and Safety of Technical Processes*, 16, 308–320.

Jia, F., Lei, Y., Lin, J., Zhou, X., and Lu, N. (2016). Deep neural networks: A promising tool for fault characteristic mining and intelligent diagnosis of rotating machinery with massive data. *Mechanical Systems and Signal Processing*, 72-73, 303–315. doi:10.1016/j.ymssp.2015.10.025.

Kazemi, Z., Safavi, A.A., Pouresmaeeli, S., and Naseri, F. (2019). A pratical framework for implementing multivariate monitoring techniques into distributed control system. *Control Engineer in Practice*, 82, 118–129. doi:10.1016/j.conengprac.2018.10.003.

Kofi, G.A., Luo, G., and Quin, K. (2013). Multi-level data pre-processing for software defect prediction. 170 – 174. doi:10.1109/ICIII.2013.6703111.

Kourti, T. (2002). Process analysis and abnormal situation detection: From theory to practice. *IEEE Control Systems Magazine*, 22(5), 10 – 25. doi:10.1109.

library for Theano, K.D.L. and TensorFlow (2020).

Martens, J., Chattopadhya, A., and Pitassi, T. (2013). On the representational efficiency of restricted boltzmann machines. *Advances in Neural Information Processing Systems 26 (NIPS 2013)*.

Moseler, O. and Isermann, R. (2000). Application of model-based fault detection to a brushless dc motor. *IEEE Transactions on industrial electronics*, 47(5), 1015–1020.

NASA (2008). That sinking feeling. system failure case studies. *National Aeronautics and Space Administration*, 2. Available at: `http://<http://nsc.nasa.gov/SFCS/SystemFailureCaseStudy/Details/11`.

Nashalji, M.N., Seyedamin, A., and Mohammad, N. (2014). Integration of principal component analysis and neural classifier for fault detection and diagnosis of tennessee eastman process. *Engineering Technology and Technopreneuship (ICE2T)*, 166–170.

Ravanbakhsh, S., Poczos, B., Schneider, J., Schuurmans, D., and Greiner, R. (2016). Stochastic neural networks with monotonic activation functions. *19th International Conference on Artificial Intelligence and Statistics - AISTATS*, 1–61.

Razvan, P., Tomas, M., and Yoshua, B. (2013). On the difficulty of training recurrent neural networks. *30th International Conference on Machine Learning*, 28.

Russell, J.S. and Norvig, P. (2003). Artificial intelligence a modern approach. Pearson Education, Inc.

Sarnovsky, M., Bednar, P., and Smatana, M. (2018). Big data processing and analytics platform architecture for process industries factories. *Big data and cognitive computing*. doi:10.3390/bdcc2010003.

Schneider, J.F., Pauwels, P., and Steiger, S. (2017). Ontology-based modeling of control logic in building automation systems. *IEEE Transactions on Industrial Informatics*, 13, 3350–3360.

Sepp, H. and Jürgen, S. (1997). Long short-term memory. *Neural Computation*, 8, 1735–1780.

Silva, M.J., Pereira, C.E., and Gotz, M. (2018). Context-aware recommendation for industrial alarm system. *16th IFAC Symposium on Information Control Problems in Manufacturing (INCOM 2018)*, 51, 229–234. doi:10.1016/j.ifacol.2018.08.266.

Silva, M.J., Pereira, C.E., and Gotz, M. (2019). On the use of inductive biases for semantic characterization of industrial alarm systems. *IEEE PES Innovative Smart Grid Technologies Conference - Latin America (ISGT Latin America)*, 229–234. doi:10.1109/ISGT-LA.2019.8894963.

Soares, V.B., Pinto, J.C., and Souza, M.B. (2016). Alarm management practices in natural gas processing plant. *Control Engineer in Practice*, 55, 185–196. doi:10.1016/j.conengprac.2016.07.004.

Su, Q., Liau, X., and Carin, L. (2017). A probabilistic framework for nonlinearities in stochastic neural networks. *31st Conference on Neural Information Processing System - NIPS 2017*, 1–10. Https://arXiv:1709.06123.

Tian, Y., Zhang, K., Lin, X., and Yang, B. (2018). Lstm-based traffic flow prediction with missing data. *Neurocomputing*, 318, 297–305. doi:10.1016/j.conengprac.2018.08.067.

Varga, T., Szeifert, F., and Abonyi, J. (2010). Detection of safe operating regions: A novel dynamic process simulator based predictive alarm management approach. *American Chemical Society*, 49, 658–668. doi:10.1021/ie9005222.

Venkatasubramanian, V., Rengaswamy, R., Kavuri, S.N., and Yin, K. (2003). A review of process fault detection and diagnosis: Part iii: Process history based methods. *Computers & chemical engineering*, 27(3), 327–346.

Wang, J., Iang, Z., Chen, K., and Zhou, D. (2017). Practices of detecting and removing nuisance alarms for alarm overloading in thermal power plants. *Control Engineering Practice*, 67, 21–30. doi:10.1016/j.conengprac.2017.07.003.

Wang, J., Yang, F., Chen, T., and Shah, s. (2016). An overview of industrial alarm systems: Main causes for alarm overloading, research status, and open problems. *Transactions on Automation Science and Engineering.*, 13, 1045–1061. doi:doi:10.1109/TASE.2015.2464234.

Yang, B., Sun, S., Li, J., Lin, X., and Tian, Y. (2019). Traffic flow prediction using lstm with feature enhancement. *Neurocomputing*, 332, 320–327. doi:10.1016/j.neucom.2018.12.016.

Yin, S., Ding, S., Haghani, A., Adel, H., Haiyang, H., and Ping, Z. (2012). A comparison study of basic data-driven fault diagnosis and process monitoring methods on the benchmark tennessee eastman process. *Control Engineering Practice*, 22, 1567–1581.

Yu, Y., Zhu, D., Wang, J., and Zhao, Y. (2017). Abnormal data detection for multivariate alarm system detection based on correlation directions. *Journal of Loss Prevention in the Process Industries*, 45, 43–55. doi:10.1016/j.jlp.2016.11.011.

Zaytonn, J. and Lafortune, S. (2013). Overview for fault diagnosis methods for discrete event systems. *Anual Reviews in Control*, 37, 308–320. doi:10.1109/ACCESS.2017.2756872.