Hybrid LSTM-Fuzzy System to Model a Sulfur Recovery Unit

Jorge S. S. Júnior[1] [a], Jérôme Mendes[2] [b], Francisco Souza[3] [c] and Cristiano Premebida[1] [d]

[1] *University of Coimbra, Institute of Systems and Robotics, DEEC, Coimbra, Portugal*

[2] *University of Coimbra, CEMMPRE, ARISE, Department of Mechanical Engineering, Coimbra, Portugal*

[3] *Radboud University, Dept. Analytical Chemistry & Chemometrics, 6525 AJ Nijmegen, The Netherlands*

Keywords: Neo-Fuzzy Neuron System, Long Short-Term Memory, Industrial Processes, Interpretability.

Abstract: Dealing with the dynamics of an industrial process using machine learning techniques has been a paradigm throughout decades of technological advancement. Motivated by addressing this problem, the present work proposes the hybridization of a neo-fuzzy neuron system (NFN) with a long short-term memory network (LSTM), the NFN-LSTM model. The fuzzy part guarantees interpretability through linguistic terms associated with membership functions that allow an effective mapping of the input variables in its universe of discourse with respect to the output. On the other hand, the LSTM part explores high-level representations useful for sequential data in dynamic processes. In this work, a sulfur recovery unit is used as a case study, whose dynamics are mainly associated with peak values in the estimation of residual hydrogen sulfide. The proposed NFN-LSTM model is compared with state-of-the-art methods, such as standalone LSTM, GAM-ZOTS (generalized additive models using zero-order Takagi-Sugeno fuzzy system), iMU-ZOTS (extension of GAM-ZOTS), ALMMo-1 (autonomous learning of a multimodel system from streaming data), iNoMO-TS (iterative learning of multivariate fuzzy models using novelty detection), and SVR (support vector regression). Analyzing the results, the proposed model performed similarly to standalone LSTM, and both outperformed the other methods. Finally, NFN-LSTM manages to balance interpretability and accuracy.

1 INTRODUCTION

The reliability and explainability of artificial intelligence (AI) models applied to modeling industrial processes are pivotal to increasing the adoption of AI in Industry 4.0 (Souza et al., 2022). The advent of complexity and nonlinearity in modern industrial systems requires computational technologies to grow at the same pace. However, the rapid progress of AI has led to very complex models, the majority of which are difficult to understand and explain. As a result, these models lack a connection with human operators, making it challenging to support reasoning or enhance understanding of the process.

It is common to see Deep Learning (DL) solutions widespread in modeling complex nonlinear processes, mainly due to the capacity to achieve high levels of data representation. For example, in (Yuan et al., 2020) the authors employed a long short-term memory network (LSTM) to learn quality-relevant hidden dynamics of a penicillin fermentation process and a debutanizer column. The study in (Guo et al., 2021) proposes a soft sensor based on the denoising autoencoder (DAE) and mechanism-introduced gated recurrent units (MGRUs) whose performance is validated by predicting the rotor thermal deformation of a rotary air preheater. Although the two DL models follow the dynamic trend of the outputs with good accuracy (especially LSTMs), they can present a large number of parameters and complexity that harm a clear understanding of the extracted features and the internal mechanism of the model (interpretability issues) (Jiang et al., 2021).

Traditional machine learning techniques can also be appreciated to model complex nonlinear processes in a more interpretable way than DL. As a motivation to reach interpretable models, an approach called GAM-ZOTS was proposed in (Mendes et al., 2019a) to learn univariate zero-order Takagi-Sugeno (T-S) fuzzy models through a backfitting algorithm. Then, an extension of the GAM-ZOTS was proposed in (Mendes et al., 2019b), the iMU-ZOTS, where

[a] https://orcid.org/0000-0001-6263-3602
[b] https://orcid.org/0000-0003-4616-3473
[c] https://orcid.org/0000-0001-6362-9349
[d] https://orcid.org/0000-0002-2168-2077

S. S. Júnior, J., Mendes, J., Souza, F. and Premebida, C.
Hybrid LSTM-Fuzzy System to Model a Sulfur Recovery Unit.
DOI: 10.5220/0012165100003543
In *Proceedings of the 20th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2023) - Volume 2*, pages 281-288
ISBN: 978-989-758-670-5; ISSN: 2184-2809

the model learns iteratively, *i.e.*, new fuzzy rules are added using a novelty detection criterion which detects if new data is not represented by the current model.

The union of deep and traditional techniques can be advantageous in exploring a balance between accuracy and interpretability, resulting in a more reliable hybrid approach for application in sensitive areas such as industrial processes. The use of fuzzy systems as a foundation to treat the interpretability side exploits the universe of discourse of the input data, easy notation readability, and a symbolic structure with comprehensible mapping of linguistic variables to fuzzy sets, despite suffering from accuracy in many complex problems (Moral et al., 2021). The survey in (Júnior et al., 2023) shows, through examples in the literature, that the combination of fuzzy systems and deep learning (the deep fuzzy systems) allows an effective trade-off between interpretability-accuracy when dealing with the individual limitations of DL and FS. This survey also shows the lack of appropriate discussions in the literature about these hybrid models, as there is no consensus on how to quantify or qualify the interpretability with potential fine adjustments from the initial design of the models.

The present work proposes a neo-fuzzy neuron architecture hybridized with LSTM, the NFN-LSTM model, for regression problems in industrial processes. The proposed model contributes to reaching an effective interpretability-accuracy trade-off by combining the good interpretability of the NFN model and the accuracy of the LSTM. The fuzzy part of the proposed model allows a mapping of the input-output relations with the easy readability of what happens globally in the system. On the other hand, the LSTM part transforms the fuzzified variables to reach a high level of representation that can follow the system's dynamics and express good accuracy. The case study chosen for this work is the estimation of residual hydrogen sulfide in the tail stream of a sulfur recovery unit, whose dynamics are perceived partly through manual control of the gas and air flows by human operators and partly through a closed-loop algorithm for airflow.

The rest of the paper is structured as follows: Section 2 presents the topics that underlie this study. Then, the proposed hybrid approach of neo-fuzzy neuron with LSTM is presented in Section 3. The implementation and validation of the proposed NFN-LSTM model are presented in Section 4. Finally, Section 5 concludes the paper.

2 BACKGROUND

In this section, an overview of topics related to the hybrid approach proposed in Section 3 will be presented, namely neo-fuzzy neuron (Section 2.1) and Long Short-Term Memory networks (Section 2.2).

2.1 The Neo-Fuzzy Neuron Model

A neo-fuzzy neuron (NFN) system incorporates multiple univariate additive zero-order Takagi-Sugeno (T-S) fuzzy systems represented by the following univariate fuzzy rules (Yamakawa, 1992):

$$R_j^{i_j} : \textbf{IF } x_j(k) \sim A_j^{i_j} \textbf{ THEN } y_j^{i_j}(k) = \theta_j^{i_j}, \quad (1)$$

where $\mathbf{x}(k) = [x_1(k), \cdots, x_n(k)]$ are input variables for the $k-$th sample ($k = 1, \cdots, K$), $R_j^{i_j}$ ($i_j = 1, \cdots, N_j$; $j = 1, \cdots, n$) depicts the i_j-th rule of j-th variable (for a total of N_j individual rules). The antecedent part is defined by the linguistic terms $A_j^{i_j}$ through fuzzy membership functions (MFs) $\mu_{A_j^{i_j}}$. The consequent part is defined by $\theta_j^{i_j}$ parameters that express the output of the univariate model $y_j^{i_j}$. Figure 1 exhibits the diagram of the described NFN system.

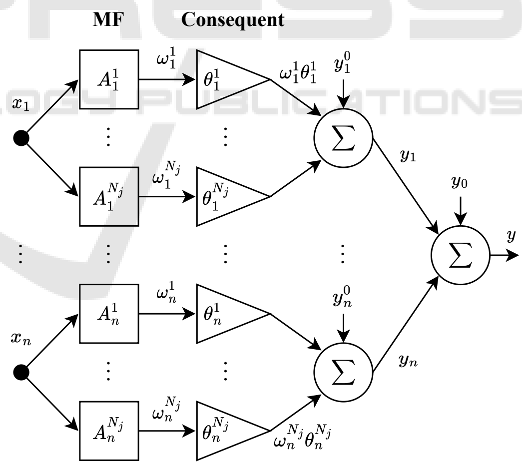

Figure 1: Neo-fuzzy neuron system.

The output of a NFN comprises the sum of the additive univariate models, where each model represents an input variable $x_{j|j=1,\cdots,n}$:

$$y[\mathbf{x}(k)] = y_0 + \sum_{j=1}^{n} y_j[x_j(k)], \quad (2)$$

with

$$y_j[x_j(k)] = y_j^0 + \sum_{i_j=1}^{N_j} \omega_j^{i_j}[x_j(k)]\theta_j^{i_j}, \quad (3)$$

where y_0 is the bias of the global model, y_j^0 is the bias of the individual model $y_j[x_j(k)]$ $(j = 1, \cdots, n)$, $\omega_j^{i_j}[x_j(k)]$ is the normalized form of the membership function $\mu_{A_j^{i_j}}$. Learning an NFN is often achieved using the backfitting algorithm, and the Least Squares Method.

2.2 The Long Short-Term Memory Networks

Long Short-Term Memory networks (LSTMs) were developed in 1997 and are mainly used in modeling tasks that use long temporal sequences (Hochreiter and Schmidhuber, 1997; Goodfellow et al., 2016). Sequential data in the time domain are processed through nonlinear elements called gates. These gates are commonly activated using sigmoidal functions (deal with irrelevant inputs and irrelevant memory contents) and hyperbolic tangent functions (avoid vanishing/exploding gradients) (Zarzycki and Ławryńczuk, 2021).

The main formulations computed at instant k within an LSTM with the structure illustrated in Figure 2 are (Zarzycki and Ławryńczuk, 2021):

$$\mathbf{i}(k) = \sigma\left(\mathbf{W}_i^{in}\mathbf{x}(k) + \mathbf{W}_i^h\mathbf{h}(k-1) + \mathbf{b}_i\right), \quad (4)$$

$$\mathbf{f}(k) = \sigma\left(\mathbf{W}_f^{in}\mathbf{x}(k) + \mathbf{W}_f^h\mathbf{h}(k-1) + \mathbf{b}_f\right), \quad (5)$$

$$\mathbf{g}(k) = \tanh\left(\mathbf{W}_g^{in}\mathbf{x}(k) + \mathbf{W}_g^h\mathbf{h}(k-1) + \mathbf{b}_g\right), \quad (6)$$

$$\mathbf{o}(k) = \sigma\left(\mathbf{W}_o^{in}\mathbf{x}(k) + \mathbf{W}_o^h\mathbf{h}(k-1) + \mathbf{b}_o\right), \quad (7)$$

$$\mathbf{s}(k) = \mathbf{f}(k) \odot \mathbf{s}(k-1) + \mathbf{i}(k) \odot \mathbf{g}(k), \quad (8)$$

$$\mathbf{h}(k) = \mathbf{o}(k) \odot \tanh(\mathbf{s}(k)), \quad (9)$$

where $\mathbf{i}(k)$, $\mathbf{f}(k)$, $\mathbf{g}(k)$ and $\mathbf{o}(k)$ are input, forget, state candidate and output gates, respectively, with respective recurrent weight matrices $\mathbf{W}_i^h, \mathbf{W}_f^h, \mathbf{W}_g^h, \mathbf{W}_o^h$, input weight matrices $\mathbf{W}_i^{in}, \mathbf{W}_f^{in}, \mathbf{W}_g^{in}, \mathbf{W}_o^{in}$, and biases $\mathbf{b}_i, \mathbf{b}_f, \mathbf{b}_g, \mathbf{b}_o$. The variables $\mathbf{x}(k)$, $\mathbf{s}(k)$ and $\mathbf{h}(k)$ represent, respectively, the input, cell state and hidden (output) vectors at k-th instant. The symbol $\odot$ is the Hadamard (element-wise) product of the vectors.

A standalone LSTM has an internal mechanism that is difficult to interpret because it has a high number of parameters required for the ports, in addition to making direct input-output mapping difficult to simplify its structure without compromising performance (Lees et al., 2022). One of the techniques used in the literature to overcome the interpretability issue is the attention mechanism that prioritizes the importance of input features (Gandin et al., 2021; Liu et al., 2021).

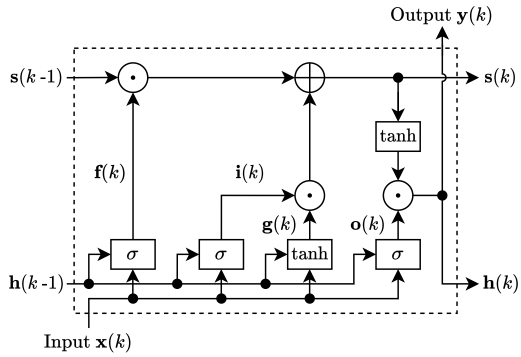

Figure 2: Representation of the Long Short-Term Memory (Júnior et al., 2023).

3 PROPOSED NFN-LSTM MODEL

3.1 Motivation

The need to develop a model that expresses reliability in its internal functioning comes partly from exploring interpretability, which can be addressed by fuzzy systems whose membership functions convey the individual impact of inputs on output. Reliability is also achieved when the model to be developed can efficiently estimate characteristics of a given system, such as dynamics and nonlinearity, with a high level of accuracy. In this context, LSTMs can be used for achieving such characteristics in systems represented by temporal sequences, as stated in Section 2.2.

The literature on eXplainable Artificial Intelligence (XAI) shows the difficulty in guaranteeing mutual excellence between interpretability and accuracy, therefore using the concept of a trade-off between both that allows a balancing with potential to be increasingly explored in the future (Angelov et al., 2021; Júnior et al., 2023). Given this concept, the present work proposes a new model that considers the interpretability capacity of fuzzy systems and the accuracy capacity of LSTMs in a hybrid structure.

3.2 NFN-LSTM Model

In this study, a LSTM network is implemented in a hybrid way to the neo-fuzzy neuron system to reach the representation portrayed in Figure 3, the NFN-LSTM model. The resulting fuzzy rules of the proposed NFN-LSTM model are expressed as:

$$R^i : \textbf{IF } x_j(k) \sim A_j^{i_j} \textbf{ THEN } y^i(k) = \text{LSTM}^i \cdot \theta^i, \quad (10)$$

where $R^i (i = 1, \cdots, N)$ represents the i-th fuzzy rule for a total of N global rules. These rules consider in ascending order each i_j-th MF $(i_j = 1, \cdots, N_j)$ of

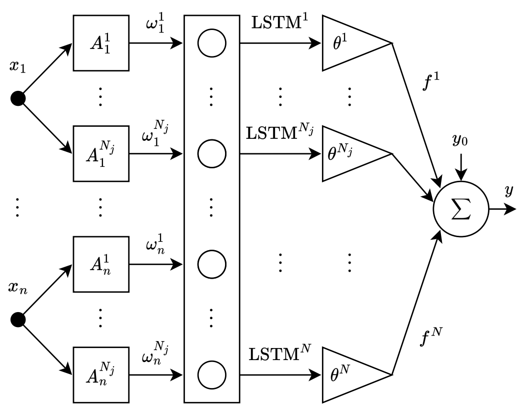

Figure 3: Representation of the NFN-LSTM model.

each $j-$th variable ($j = 1, \cdots, n$) according to Figure 3 (therefore $N = n \cdot N_j$). LSTM outputs are represented by parameters LSTM^i that accompany the consequent parameters θ^i. The MFs used in the hybrid model are of the complementary triangular form that can be activated at the instant k in pairs (at most, for a given input x_j), i.e. (Silva et al., 2014):

$$\sum_{i_j=1}^{N_j} \mu_{A_j^{i_j}}[x_j(k)] = \mu_{A_j^{i_j}}[x_j(k)] + \mu_{A_j^{i_j+1}}[x_j(k)] = 1. \quad (11)$$

The output of the hybrid model is expressed as:

$$y[\mathbf{x}(k)] = y_0 + \sum_{i=1}^{N} \omega^i f^i[x_{j|j=1,\cdots,n}(k)], \quad (12)$$

where y_0 is a bias and

$$\left[\omega^1, \cdots, \omega^N\right] \equiv \left[\omega_1^1, \cdots, \omega_n^{N_j}\right] \quad (13)$$

$$\omega_j^{i_j} \equiv \omega_j^{i_j}[x_j(k)] = \frac{\mu_{A_j^{i_j}}[x_j(k)]}{\sum_{i_j=1}^{N_j} \mu_{A_j^{i_j}}[x_j(k)]} \quad (14)$$

$$f^i[x_{j|j=1,\cdots,n}(k)] = \text{LSTM}^i(\omega_1^1, \cdots, \omega_n^{N_j}) \cdot \theta^i \quad (15)$$

As noted in Section 2.2 and Figure 3, it becomes an arduous task to try to establish a direct relationship between the inputs $\omega_j^{i_j}$ and outputs LSTM^i of the LSTM due to the high complexity of its internal mechanism. For this reason, the outputs LSTM^i described in Eq. (15) are defined as dependent variables of all inputs $\omega_1^1, \cdots, \omega_n^{N_j}$.

4 RESULTS

This section deals with the implementation of the NFN-LSTM proposed in this work. Section 4.1 defines the case study and the choice of state-of-the-art methods to compare with NFN-LSTM. Next, the experimental results of the methods analyzed in Section 4.2 are presented and discussed.

Input: Input variables
$\mathbf{x} : \{(x_1(k), \cdots, x_n(k))\}_{k=1}^K$, each one with N_j MFs, time steps, LSTM layers
begin
Compute triangular MFs based on Eq. (11);
Normalize MFs and obtain $\left[\omega^1, \cdots, \omega^N\right]$ according to Eq. (13) and Eq. (14);
Compute LSTM$^i(\omega^1, \cdots, \omega^N)$ (LSTM outputs) based on Eq. (4)-(9);
Obtain the consequent parameters $f^i[x_{j|j=1,\cdots,n}(k)]$ using Eq. (15);
Estimate the final output using Eq. (12);
end
Output: Estimated output $\hat{\mathbf{y}} : \{y(k)\}_{k=1}^K$

Algorithm 1: NFN-LSTM learning.

4.1 Case Study

The model proposed in this paper was implemented using an industrial case study that deals with the estimation of residual hydrogen sulfide (H_2S) in the tail stream of a Sulfur Recovery Unit (SRU) (Fortuna et al., 2007). SRU has four sulfur lines that operate in parallel through two separate combustion chambers, where one is fed with MEA gas (from the gas washing plants) rich in H_2S and the other is fed with SWS gas (from sour water stripping plants) rich in H_2S and ammonia (NH_3) (Fortuna et al., 2003). Correct estimation of residual H_2S is understood as having the ability to detect peaks in their values as they indicate undesirable behavior of the SRU (Souza et al., 2022).

To estimate the residual H_2S, five variables are used (Curreri et al., 2021):

1. $\mathbf{x}_1$: MEA gas fed to the first combustion chamber;
2. $\mathbf{x}_2$: air fed to the first combustion chamber, that regulates the combustion of MEA gas to supply oxygen for the reaction;
3. $\mathbf{x}_3$: automatically controlled air flow to improve the stoichiometric ratio $[H_2S] - 2[SO_2]$;
4. $\mathbf{x}_4$: total gas fed to the second combustion chamber, composed of SWS gas and additional MEA gas (required when SWS gas is too low);
5. $\mathbf{x}_5$: total air fed to the second combustion chamber, composed of air flow for the combustion of SWS gas and an additional air flow to keep gas input constant.

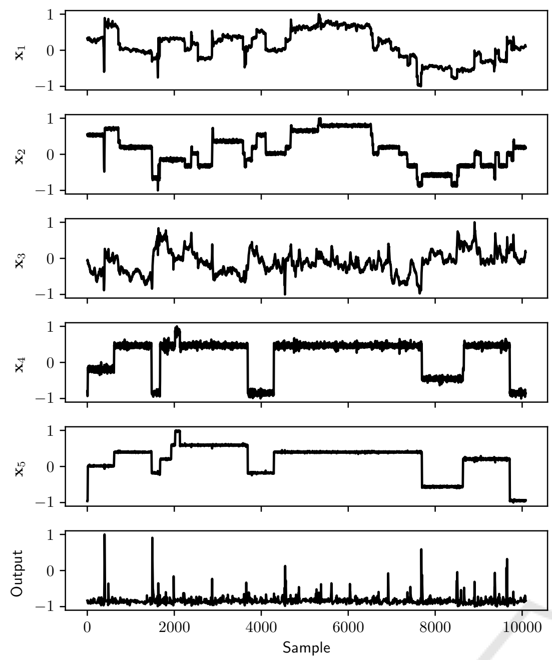

Figure 4: Input and output data from SRU.

Figure 4 shows the dataset used to implement the proposed model, which has $\mathbf{x}_1, \cdots, \mathbf{x}_5$ as inputs and the residual H_2S as output, each containing $K = 10081$ samples. This dataset was split into 60% training, 10% validation, and 30% testing (*i.e.*, 6094, 1008, and 3024 samples, respectively).

The proposed NFN-LSTM is validated through comparison with the following methods:

- LSTM: standalone long short-term memory network (Hochreiter and Schmidhuber, 1997);
- GAM-ZOTS: generalized additive models using zero-order T-S fuzzy systems learned by backfitting algorithm (Mendes et al., 2019a);
- iMU-ZOTS: iterative learning for a model composed of the sum of multiple univariate zero-order T-S fuzzy systems (Mendes et al., 2019b);
- ALMMo-1: autonomous learning of a multimodel system from streaming data (first-order predictor) (Angelov et al., 2018);
- iNoMO-TS: iterative learning of multi-input multi-output T-S fuzzy models using novelty detection (Júnior et al., 2021);
- SVR: support vector regression (Vapnik, 1999).

The training and testing datasets are used for learning the presented models. The validation dataset is exclusively used in the hybrid model to individually verify the model's tendency to behave with unknown data throughout training in several epochs.

The performance of the proposed model is validated using Normalized Root Mean Square Error (NRMSE) and Mean Absolute Error (MAE):

$$NRMSE = \frac{\sqrt{\frac{\sum_{k=1}^{K}[y(k)-\hat{y}(k)]^2}{K}}}{\max(\mathbf{y})-\min(\mathbf{y})}, \quad (16)$$

$$MAE = \frac{\sum_{k=1}^{K}|y(k)-\hat{y}(k)|}{K}, \quad (17)$$

where K is the total number of samples, $y(k)$ and $\hat{y}(k)$ are the real and estimated outputs of the system at k-th sample, respectively, and $\max(\mathbf{y})$ and $\min(\mathbf{y})$ are the maximum and minimum values of output $\mathbf{y} = [y(1), \cdots, y(K)]$. The smaller the NRMSE and MAE values (closer to zero), the better the performance of the method to be validated.

Experiments with NFN-LSTM, LSTM, ALMMo-1, iNoMO-TS, and SVR were performed using Spyder 5 (Python, PyTorch). On the other hand, experiments with GAM-ZOTS and iMU-ZOTS were performed using MATLAB 2022a and C language. Each one of the experiments was performed 10 times on an AMD Ryzen 4800H CPU @ 2.90 GHz, with 16GB DDR4 and 512GB PCIe SSD.

4.2 Implementation

The proposed model has been implemented with 5 membership functions ($N_j = 5$) for each j−th variable ($j = 1, \cdots, 5$), being the total number of fuzzy rules $N = n \cdot N_j = 5 \cdot 5 = 25$ as the dimension of the input and output of the LSTM. The part with LSTM has only one hidden layer and one-step forecasting. The learning process uses the Adam algorithm (Kingma and Ba, 2014) as optimizer, mean squared error (MSE) as loss function, 100 epochs, and learning rate 10^{-1} decaying stepwise by 0.95 every five epochs.

Figure 5 shows the losses for the 100 epochs of the best trial (out of 10). The comparison results between the real and estimated outputs in training and testing stages of the best trial are shown in Figure 6. The training error values obtained were $NRMSE = 0.0348$ and $MAE = 0.0422$, while the testing error values were $NRMSE = 0.0401$ and $MAE = 0.0417$.

In Figure 5, the training and validation cost functions converged satisfactorily, indicating that the proposed model responds well to unknown data. Figure 6 shows that the model reached some peaks as expected from the SRU dataset, although there are still opportunities to improve performance and address more peaks concerning the system's dynamic. The fuzzy rule base obtained from the training of the NFN-

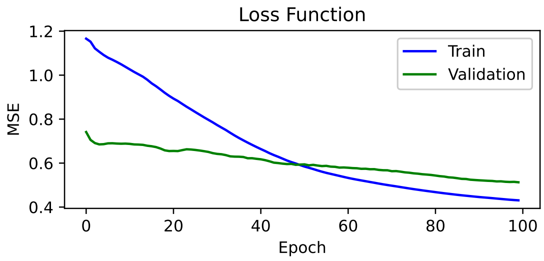

Figure 5: Loss functions for training and validation.

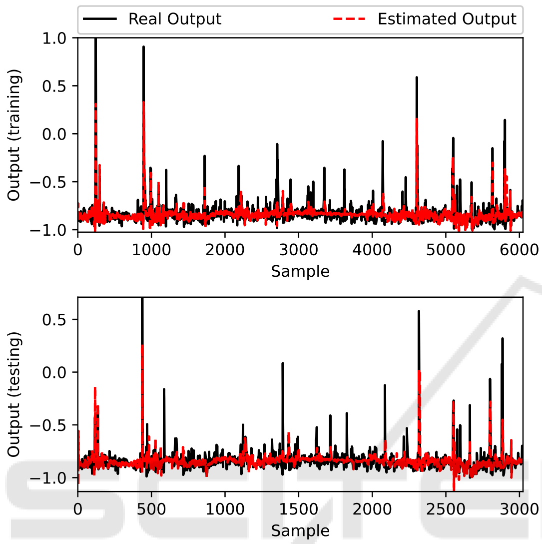

Figure 6: Results of the proposed NFN-LSTM model for estimating H_2S residual in the tail stream of SRU.

LSTM model (best trial), in the form of Eq. (10), is represented below.

Fuzzy rules for $x_1(k)$:

$$\textbf{IF } x_1(k) \sim A_1^1 \textbf{ THEN } y^1(k) = 5.18 \cdot \text{LSTM}^1$$
$$\textbf{IF } x_1(k) \sim A_1^2 \textbf{ THEN } y^2(k) = 5.03 \cdot \text{LSTM}^2$$
$$\textbf{IF } x_1(k) \sim A_1^3 \textbf{ THEN } y^3(k) = 5.47 \cdot \text{LSTM}^3$$
$$\textbf{IF } x_1(k) \sim A_1^4 \textbf{ THEN } y^4(k) = 4.85 \cdot \text{LSTM}^4$$
$$\textbf{IF } x_1(k) \sim A_1^5 \textbf{ THEN } y^5(k) = 6.66 \cdot \text{LSTM}^5$$

Fuzzy rules for $x_2(k)$:

$$\textbf{IF } x_2(k) \sim A_2^1 \textbf{ THEN } y^6(k) = 5.27 \cdot \text{LSTM}^6$$
$$\textbf{IF } x_2(k) \sim A_2^2 \textbf{ THEN } y^7(k) = 4.50 \cdot \text{LSTM}^7$$
$$\textbf{IF } x_2(k) \sim A_2^3 \textbf{ THEN } y^8(k) = 5.03 \cdot \text{LSTM}^8$$
$$\textbf{IF } x_2(k) \sim A_2^4 \textbf{ THEN } y^9(k) = 3.43 \cdot \text{LSTM}^9$$
$$\textbf{IF } x_2(k) \sim A_2^5 \textbf{ THEN } y^{10}(k) = -0.03 \cdot \text{LSTM}^{10}$$

Fuzzy rules for $x_3(k)$:

$$\textbf{IF } x_3(k) \sim A_3^1 \textbf{ THEN } y^{11}(k) = 2.11 \cdot \text{LSTM}^{11}$$
$$\textbf{IF } x_3(k) \sim A_3^2 \textbf{ THEN } y^{12}(k) = -0.14 \cdot \text{LSTM}^{12}$$
$$\textbf{IF } x_3(k) \sim A_3^3 \textbf{ THEN } y^{13}(k) = 0.21 \cdot \text{LSTM}^{13}$$
$$\textbf{IF } x_3(k) \sim A_3^4 \textbf{ THEN } y^{14}(k) = 3.62 \cdot \text{LSTM}^{14}$$
$$\textbf{IF } x_3(k) \sim A_3^5 \textbf{ THEN } y^{15}(k) = 2.24 \cdot \text{LSTM}^{15}$$

Fuzzy rules for $x_4(k)$:

$$\textbf{IF } x_4(k) \sim A_4^1 \textbf{ THEN } y^{16}(k) = 3.92 \cdot \text{LSTM}^{16}$$
$$\textbf{IF } x_4(k) \sim A_4^2 \textbf{ THEN } y^{17}(k) = 3.94 \cdot \text{LSTM}^{17}$$
$$\textbf{IF } x_4(k) \sim A_4^3 \textbf{ THEN } y^{18}(k) = 5.58 \cdot \text{LSTM}^{18}$$
$$\textbf{IF } x_4(k) \sim A_4^4 \textbf{ THEN } y^{19}(k) = -0.55 \cdot \text{LSTM}^{19}$$
$$\textbf{IF } x_4(k) \sim A_4^5 \textbf{ THEN } y^{20}(k) = 5.00 \cdot \text{LSTM}^{20}$$

Fuzzy rules for $x_5(k)$:

$$\textbf{IF } x_5(k) \sim A_5^1 \textbf{ THEN } y^{21}(k) = 4.00 \cdot \text{LSTM}^{21}$$
$$\textbf{IF } x_5(k) \sim A_5^2 \textbf{ THEN } y^{22}(k) = 2.56 \cdot \text{LSTM}^{22}$$
$$\textbf{IF } x_5(k) \sim A_5^3 \textbf{ THEN } y^{23}(k) = 0.05 \cdot \text{LSTM}^{23}$$
$$\textbf{IF } x_5(k) \sim A_5^4 \textbf{ THEN } y^{24}(k) = -2.52 \cdot \text{LSTM}^{24}$$
$$\textbf{IF } x_5(k) \sim A_5^5 \textbf{ THEN } y^{25}(k) = -0.17 \cdot \text{LSTM}^{25}$$

4.3 Discussion

Different methods may present distinct behaviors to attempt a fair comparison of performance. Thus, the main parameters of the methods compared in this section were adjusted as follows:

- LSTM: one hidden layer, one-step forecasting, 25 for input and output dimension, 100 epochs, learning rate 10^{-1} decaying stepwise by 0.95 every five epochs, Adam optimizer.
- GAM-ZOTS: $it_{max} = 100$ (max. number of iterations), $\xi = 10^{-3}$ (termination condition), $N_j = 5$ (fixed number of rules), backfitting algorithm.
- iMU-ZOTS: $it_{max} = 100$ (max. number of iterations), $\xi = 10^{-3}$ (termination condition), $N_j = 5$ (fixed max. number of rules), $M_{th}^j = 0.8$ (threshold for novelty detection), backfitting algorithm.
- ALMMo-1: $\Omega = 10$ (for initializing covariance matrices), $\lambda = 0.8$ (threshold for adding new rules), $\eta = 0.1$ (forgetting factor).
- iNoMO-TS: $it_{max} = 100$ (max. number of iterations), $N_D^{th} = 0.3$ (threshold for novelty detection), $S_D^{th} = 0.6$ (threshold for similarity detection), $\sigma_{ini} = 10$ (elements of the initial covariance matrix).
- SVR: $C = 1$ (regularization parameter), epsilon = 0.1 (threshold for non-penalty associated with errors), radial basis function kernel.

Table 1: Testing results of the proposed NFN-LSTM, LSTM, GAM-ZOTS, iMU-ZOTS, ALMMo-1, iNoMO-TS, and SVR.

Method	Elapsed (s)	NRMSE	MAE
NFN-LSTM	2.7672	**0.0414**	**0.0425**
LSTM	2.4712	0.0436	0.0452
GAM-ZOTS	4.7521	0.0542	0.0509
iMU-ZOTS	1.8751	0.0529	0.0518
ALMMo-1	13.6527	0.0649	0.0794
iNoMO-TS	353.5502	0.2003	0.2332
SVR	**1.3709**	0.0510	0.0428

Table 1 presents the average elapsed training time and the average testing error values of 10 trials of the proposed model, LSTM, GAM-ZOTS, iMU-ZOTS, ALMMo-1, iNoMO-TS, and SVR, whose best average performance is highlighted in bold. Figure 7 compares the outputs of the best-learned models across 10 trials in the testing phase.

It can be noticed that the proposed model and the LSTM had similar performance in estimation quality. This fact does not occur in the other methods, especially GAM-ZOTS and iMU-ZOTS, which have a neo-fuzzy neuron model with a similar structure that provided the basis for the proposed method. Furthermore, ALMMo-1 and iNoMO-TS did not obtain efficient estimation despite their adaptive learning, whose final number of membership functions for each input variable (as well as fuzzy rules) was equal to 12 and 15, respectively. Still, Figure 7 shows the instability expressed by iNoMO-TS and the low estimation quality by SVR, although it presents error values close to the NFN-LSTM and LSTM. In addition to achieving good accuracy and estimation quality, the proposed model (NFN-LSTM) manages to express good interpretability in a global way that can be analyzed on the learned fuzzy rules.

5 CONCLUSIONS

The present study proposed a hybrid method involving a neo-fuzzy neuron system with long short-term memory, the NFN-LSTM model, with a positive balance between interpretability and accuracy. Interpretability can be analyzed by the influence of inputs on the output with the learned fuzzy rules, and accuracy is verified by the quality of the output estimation and associated errors. The case study in an industrial system showed the viability and power of NFN-LSTM over the LSTM, GAM-ZOTS, iMU-ZOTS, ALMMo-1, iNoMO-TS, and SVR methods, due to the results and the additional task of detecting important peak values for the application. Fu-

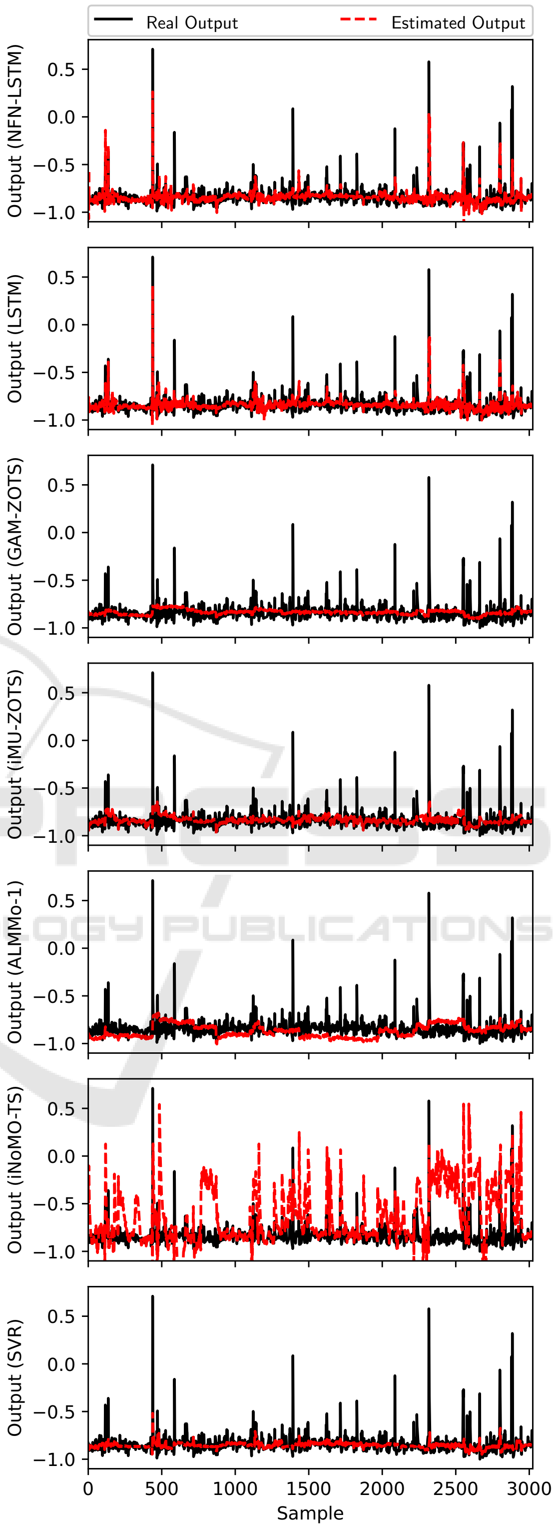

Figure 7: Comparison of testing outputs of the NFN-LSTM, LSTM, GAM-ZOTS, iMU-ZOTS, ALMMo-1, iNoMO-TS, and SVR.

ture works will explore the use of other methods in a hybrid way to the neo-fuzzy neuron to improve the interpretability-accuracy trade-off.

ACKNOWLEDGEMENTS

Jorge S. S. Júnior is supported by Fundação para a Ciência e a Tecnologia (FCT) under the grant ref. 2021.04917.BD. This research was supported by the ERDF and national funds through the project InGestAlgae (CENTRO-01-0247-FEDER-046983), and by the FCT project UIDB/00048/2020.

REFERENCES

Angelov, P. P., Gu, X., and Príncipe, J. C. (2018). Autonomous Learning Multimodel Systems From Data Streams. *IEEE Transactions on Fuzzy Systems*, 26(4):2213–2224.

Angelov, P. P., Soares, E. A., Jiang, R., Arnold, N. I., and Atkinson, P. M. (2021). Explainable artificial intelligence: an analytical review. *WIREs Data Mining and Knowledge Discovery*.

Curreri, F., Patanè, L., and Xibilia, M. (2021). Soft Sensor Transferability between Lines of a Sulfur Recovery Unit. *IFAC-PapersOnLine*, 54(7):535–540.

Fortuna, L., Graziani, S., Rizzo, A., and Xibilia, M. G. (2007). *Soft Sensors for Monitoring and Control of Industrial Processes*. Springer Science & Business Media.

Fortuna, L., Rizzo, A., Sinatra, M., and Xibilia, M. (2003). Soft analyzers for a sulfur recovery unit. *Control Engineering Practice*, 11(12):1491–1500.

Gandin, I., Scagnetto, A., Romani, S., and Barbati, G. (2021). Interpretability of time-series deep learning models: A study in cardiovascular patients admitted to Intensive care unit. *Journal of Biomedical Informatics*, 121:103876.

Goodfellow, I., Bengio, Y., Courville, A., and Bengio, Y. (2016). *Deep learning*. MIT press Cambridge.

Guo, R., Liu, H., Wang, W., Xie, G., and Zhang, Y. (2021). A Hybrid-driven Soft Sensor with Complex Process Data Based on DAE and Mechanism-introduced GRU. In *2021 IEEE 10th Data Driven Control and Learning Systems Conference (DDCLS)*, pages 553–558.

Hochreiter, S. and Schmidhuber, J. (1997). Long Short-Term Memory. *Neural Computation*, 9(8):1735–1780.

Jiang, Y., Yin, S., Dong, J., and Kaynak, O. (2021). A Review on Soft Sensors for Monitoring, Control, and Optimization of Industrial Processes. *IEEE Sensors Journal*, 21(11):12868–12881.

Júnior, J. S. S., Mendes, J., Araújo, R., Paulo, J. R., and Premebida, C. (2021). Novelty Detection for Iterative Learning of MIMO Fuzzy Systems. In *2021 IEEE 19th International Conference on Industrial Informatics (INDIN)*, pages 1–7.

Júnior, J. S. S., Mendes, J., Souza, F., and Premebida, C. (2023). Survey on Deep Fuzzy Systems in regression applications: a view on interpretability. *International Journal of Fuzzy Systems*, pages 1–22.

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

Lees, T., Reece, S., Kratzert, F., Klotz, D., Gauch, M., Bruijn, J. D., Sahu, R. K., Greve, P., Slater, L., and Dadson, S. J. (2022). Hydrological concept formation inside long short-term memory (LSTM) networks. *Hydrology and Earth System Sciences*, 26(12):3079–3101.

Liu, X., Shi, Q., Liu, Z., and Yuan, J. (2021). Using LSTM Neural Network Based on Improved PSO and Attention Mechanism for Predicting the Effluent COD in a Wastewater Treatment Plant. *IEEE Access*, 9:146082–146096.

Mendes, J., Souza, F., Araújo, R., and Rastegar, S. (2019a). Neo-Fuzzy Neuron Learning Using Backfitting Algorithm. *Neural Computing and Applications*, 31(8):3609–3618.

Mendes, J., Souza, F. A. A., Maia, R., and Araújo, R. (2019b). Iterative Learning of Multiple Univariate Zero-Order T-S Fuzzy Systems. In *Proc. of the The IEEE 45th Annual Conference of the Industrial Electronics Society (IECON 2019)*, pages 3657–3662. IEEE.

Moral, J. M. A., Castiello, C., Magdalena, L., and Mencar, C. (2021). *Explainable Fuzzy Systems*. Springer International Publishing.

Silva, A. M., Caminhas, W., Lemos, A., and Gomide, F. (2014). A fast learning algorithm for evolving neo-fuzzy neuron. *Applied Soft Computing*, 14:194–209.

Souza, F., Offermans, T., Barendse, R., Postma, G., and Jansen, J. (2022). Contextual Mixture of Experts: Integrating Knowledge into Predictive Modeling. *IEEE Transactions on Industrial Informatics*, pages 1–12.

Vapnik, V. (1999). *The nature of statistical learning theory*. Springer science & business media.

Yamakawa, T. (1992). A Neo Fuzzy Neuron and Its Applications to System Identification and Prediction of the System Behavior. *Proc. of the 2nd Int. Conf. on Fuzzy Logic & Neural Networks*, pages 477–483.

Yuan, X., Li, L., and Wang, Y. (2020). Nonlinear Dynamic Soft Sensor Modeling With Supervised Long Short-Term Memory Network. *IEEE Transactions on Industrial Informatics*, 16(5):3168–3176.

Zarzycki, K. and Ławryńczuk, M. (2021). LSTM and GRU Neural Networks as Models of Dynamical Processes Used in Predictive Control: A Comparison of Models Developed for Two Chemical Reactors. *Sensors*, 21(16).