



Article

A Machine Learning-Based Method for Modelling a Proprietary SO₂ Removal System in the Oil and Gas Sector

Francesco Grimaccia ¹, Marco Montini ², Alessandro Nicolai ¹, Silvia Taddei ² and Silvia Trimarchi ^{1,*}¹ Department of Energy, Politecnico di Milano, 20156 Milan, Italy² Eni S.p.A., Strada Statale 9, Via Emilia 1, San Donato Milanese, 20097 Milan, Italy

* Correspondence: silvia.trimarchi@polimi.it

Abstract: The aim of this study is to develop a model for a proprietary SO₂ removal technology by using machine learning techniques and, more specifically, by exploiting the potentialities of artificial neural networks (ANNs). This technology is employed at the Eni oil and gas treatment plant in southern Italy. The amine circulating in this unit, that allows for a reduction in the SO₂ concentration in the flue gases and to be compliant with the required specifications, is a proprietary solvent; thus, its composition is not publicly available. This has led to the idea of developing a machine learning (ML) algorithm for the unit description, with the objective of becoming independent from the licensor and more flexible in unit modelling. The model was developed in MatLab[®] by implementing ANNs and the aim was to predict three targets, namely the flow rate of SO₂ that goes to the Claus unit, the emissions of SO₂, and the flow rate of steam sent to the regenerator reboiler. These represent, respectively, the two physical outputs of the unit and a proxy variable of the amine quality. Three different models were developed, one for each target, that employed the Levenberg–Marquardt optimization algorithm. In addition, the ANN topology was optimized case by case. From the analysis of the results, it emerged that with a purely data-driven technique, the targets can be predicted with good accuracy. Therefore, this model can be employed to better manage the SO₂ removal system, since it allows for the definition of an optimal control strategy and the maximization of the plant's productivity by not exceeding the process constraints.

Keywords: machine learning; neural networks; oil and gas; SO₂ removal technology



Citation: Grimaccia, F.; Montini, M.; Nicolai, A.; Taddei, S.; Trimarchi, S. A Machine Learning-Based Method for Modelling a Proprietary SO₂ Removal System in the Oil and Gas Sector. *Energies* **2022**, *15*, 9138. <https://doi.org/10.3390/en15239138>

Academic Editor: Dino Musmarra

Received: 15 October 2022

Accepted: 15 November 2022

Published: 2 December 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



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

1. Introduction

In the oil and gas industry, a digitalization process is ongoing, achieved by exploiting the potentialities of artificial intelligence (AI) and machine learning (ML). This allows for better control of the production processes and improvements in the safety of operations, energy efficiency, and environmental emissions [1,2]. Within this framework, this study aims to develop a machine learning model for a description of a proprietary sulphur dioxide removal system.

The technology, among its different applications, it is used in the flue gas treatment at the Eni oil and gas plant in southern Italy, which is the object of this study. This system reduces the SO₂ concentration in the flue gases below the limit imposed by law [3]. This system is crucial since sulphur dioxide is a toxic gas. It is colourless and irritating, soluble in water, and can cause respiratory issues and damage to the environment [4].

To drastically reduce the SO₂ concentration in gases, the technology employs an amine solvent, whose composition is however unknown [5]. Therefore, it is not possible to develop a physical simulation of the system in commercial software typically used for process simulations. As a possible alternative, ML techniques can be applied to predict the licensed process unit's output values.

With this in mind, to describe the unit's behaviour, a model was developed in MatLab R2021b[®] by implementing artificial neural networks (ANNs). The aim of this work is to

model a unit whose physical-chemical modelling is unknown. In this perspective, the scope is to predict the values of three different variables, namely the two physical outputs of the unit and a proxy variable of the amine quality, which allows us to determine how the system is performing. These are, respectively, the flow rate of SO₂ sent to the Claus unit, the emissions of SO₂ expressed as a concentration, and the flow rate of steam going to the regenerator column. This surrogate model of the proprietary system allows us to better manage the system and to find an optimal operating strategy. In particular, it allows for the maximization of the productivity of a plant by always keeping the operating point of the unit within the process constraints. This is achieved by keeping both the emissions of sulphur dioxide and the flow rate of the product flowing to another unit under control. In addition, since the model also provides insights into amine quality, it can be employed to define predictive maintenance operations and to verify which working conditions ensure a longer and higher amine quality.

Machine Learning models have a variety of applications in the oil and gas industry, for example, to find descriptions for and optimize oil and gas separation and treatment plants [6,7]. However, with regard to this segment, in the literature, it is only possible to find studies where only some of the equipment of a unit has been modelled using ML techniques and therefore, this study suggests a new method to model the proprietary units of an oil and gas treatment plant.

One of the most common and representative examples found in the literature is the study conducted by Tavakoli et al. Here, the authors aimed to predict the outputs of an absorption column of a gas-sweetening plant in Iran. These are the water content, H₂S, and CO₂ mole fraction, which represent the absorption capacity of the amine circulating in the unit. This objective was achieved by implementing feed-forward ANNs, whose architecture was optimized by exploiting genetic algorithms [8].

Similarly, Salooki et al. predicted the outputs of a sweetening regenerator column in a refinery by implementing feed-forward ANNs. In this case, the selected method showed high accuracy in the prediction of the desired targets [9]. The final objective of both these studies was to determine numerically the output values of the equipment, given the expensive and time-consuming experimental measurements otherwise required [8,9].

Another study where ANNs were implemented for predicting the behaviour of equipment in an oil and gas treatment plant was the one carried out by Kwon et al. Different ANN models were tested and the long short-term memory (LSTM) model predicted the production stage temperature of a distillation column with the greatest accuracy. This parameter was a proxy variable of the product composition and its control determined the reduction in the quantity of steam that was necessary to produce to carry out the distillation. Therefore, the energy consumed by the process was reduced and its energy efficiency increased [10].

On the other hand, Kundu et al. developed a model for the prediction of the furnace coil outlet temperature of a crude distillation column by employing the random forest algorithm. This parameter, if tuned correctly, allows for a decrease in the fuel fired in the furnace and the maximization of the distillate yield. An optimal result was achieved in the prediction of the target, with an R² higher than 0.9 [11].

Finally, Adib et al. implemented the support vector machine (SVM) method to predict the outputs of a stabilization column in a sweetening plant in Iran. These were the volatility and the H₂S concentration of the condensate, whose prediction was achieved with a coefficient higher than 0.9, and these can be used for the supervision and control of the process [12].

Starting from the examples found in the literature, this study has a wider approach since it models the whole unit of an oil and gas treatment plant. Therefore, it enables not only the better management of a whole unit by predicting its outputs and crucial parameters but also the optimization of the working operation of the entire plant.

This paper is structured as follows. Section 2 describes the working principles of the proprietary technology and the data set employed for the development of the model, that

is, the targets, the time period, and the data sampling are defined. Then, in Section 3, the machine learning method employed is explained, whereas Section 4 displays the results achieved. Each target is analyzed singularly and the performance is discussed both in terms of accuracy and computational cost. Finally, Section 5 discloses the main conclusions that can be drawn from the study and the possible further development of this work.

2. Unit Description and Data Analysis

The system in exam is a regenerable scrubbing technology that can reduce the SO₂ content in the flue gases of a plant by up to 20 ppm by employing an aminic-based solution. This study focuses the system located in the Eni plant in southern Italy [5].

This unit can be found right before the chimney of the plant and it is preceded by the quench and saturation unit, which is a pre-treatment section consisting of a quench tower and a packed column, where the acid mists and suspended solids are removed from the flue gases [13].

The system is composed of three main units: an SO₂ absorption column, an amine regeneration column, and an absorbent purification unit (APU).

As depicted in Figure 1, the gases coming from the pre-treatment section enter at the bottom of the absorption column. Here, they meet in a counter-current the lean amine, which enters the column from the top. Two streams exit the absorber, one from the top and one from the bottom. They are, respectively, the treated gas that is sent to the chimney and the amine rich in SO₂ that goes into the regeneration column. The latter is warmed up in a heat exchanger in a counter-current with the poor amine and is recirculated thanks to two pumps. The amine solution, in contact with the flue gases, absorbs up to 81% of the SO₂ originally present through chemical absorption [5,13].

Contrary to the absorption process, the regeneration process is endothermic. The column is divided into two sections, a stripping and an enrichment one. The rich amine enters at the top of the column above the stripping bed and once it has been regenerated, it exits the column from the bottom, as shown in Figure 1. Two reboilers (one running and one spare) are installed in the lower part of the column, through which the heat necessary for the regeneration process is supplied. The lean amine is then sent to the storage tank after being cooled thanks to a heat exchanger that works with the rich amine in a counter-current. On the other hand, a stream of gases and vapours rich in SO₂ leaves the column from the top. This is then condensed and the gas rich in SO₂ is separated in a reflux accumulator and sent to the Claus unit [5,13].

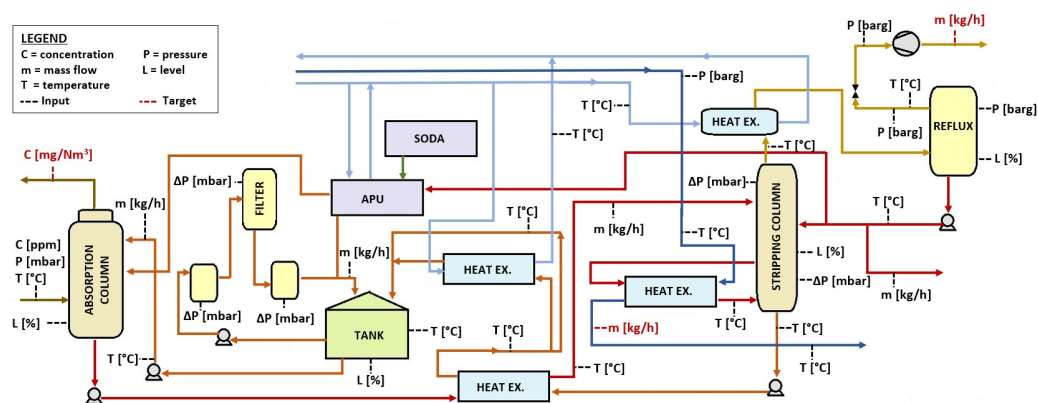


Figure 1. Scheme of SO₂ removal system located in the plant in southern Italy. The arrows represent the fluxes entering and exiting the equipment. The input variables are shown in black, whereas the targets are shown in red. Both are connected to the measurement points with dotted lines.

The Claus process aims to convert into elemental sulphur the hydrogen sulphide and the sulphur dioxide, which are two compounds always present in crude oil and natural gas. The process consists of two stages. In the first stage, the H₂S coming from the gas sweetening unit is converted to SO₂, which together with the sulphur dioxide coming from

the removal system, is converted into elemental sulphur in a second reactor. The overall conversion efficiency is usually around 96–97%, a value that is however not high enough to fulfil Italian regulations. Therefore, after the Claus unit, the unconverted gases must be sent to a tail gas treatment unit [14].

In addition, the analyzed system is composed of a storage tank for the amine, an amine purification unit (APU) for the solvent regeneration, and a filtration system to remove the suspended solids [5,15].

To build the model of the unit, all the variables available regarding the SO₂ removal system were considered as inputs. These are shown in Figure 1, corresponding to their measurement points, and mainly represent physical variables such as flow rates, pressures, temperatures, or pressure drops. On the other hand, three different targets were identified as dependent variables and their prediction enables a determination of how the system is performing. The first is the flow rate of the SO₂ leaving the regenerator column and going to the Claus unit. The second represents the flow rate of the steam entering the stripping column, which is an indicator of amine quality. The third is the concentration of SO₂ in the fumes leaving the absorber that are vented to the chimney. In this way, for the development of the ML model, 35 inputs and 3 targets were considered.

A time span of 26 months was studied, that is, from 1 June 2019 to 31 August 2021, and the data were sampled every 5 min, which is the minimum available sampling frequency. This was limited by the variable representing the SO₂ concentration at the absorber outlet, which was a KPI defined as the average over 5 min.

3. Methods

To build the model, it was necessary to employ a machine learning method that was able to represent and account for the complex nonlinear relationships between the input and output variables that occur in a chemical plant. With this in mind, and also by looking at the examples found in the literature, three different methods were identified as suitable for modelling the SO₂ removal system. These were the random forest, artificial neural network, and support vector machine (SVM) methods [16–18]. Therefore, a first-attempt analysis was carried out using the MatLab tool *RegressionLearner* in order to compare the performances of the different algorithms for the prediction of the first target, namely the flow rate of SO₂ that goes to the Claus unit. The results achieved are summarized in Table 1 and are compared in terms of the normalized root mean square error (nRMSE) on the validation and testing datasets. The expression of the nRMSE, which is equal to the RMSE divided by the maximum value of the target, is shown in Equation (1) [19,20]:

$$nRMSE = \frac{\sqrt{\sum_{i=1}^n \frac{(y_i - \hat{y}_i)^2}{n}}}{\bar{y}} \quad (1)$$

where y_i is the actual value of the target, \hat{y}_i is the predicted value, n is the total number of observations, and \bar{y} is the maximum value of the target in question.

Table 1. Comparison of the random forest, artificial neural network, and support vector machine methods for the prediction of the SO₂ flow rate. The values of the nRMSE on the validation and testing datasets are reported.

	Random Forest	Neural Networks	Support Vector Machine
nRMSE validation	0.0394	0.0359	0.049
nRMSE test	0.0383	0.0353	0.0511

It emerged that the SVM method produced the worst results in terms of the nRMSE both on the validation and testing data. On the other hand, the values achieved for the random forest and ANN methods were closer but the latter produced lower error values

and hence better results. In addition, ANNs are not only able to represent and account for complex nonlinear relationships but they are also global techniques for regression problems and are not greatly affected by the noise in the input data, which in the considered case is not negligible [16,21]. Therefore, the model of the technology was implemented in MatLab using artificial neural networks.

An ANN is composed of strict interconnections of artificial neurons and consists of three main parts: an input layer, one or more hidden layers, and an output layer. Each connection is characterized by a weight and every artificial neuron employs an activation function, which determines the neuron's output value [22].

The network learns by adjusting the weights during the training phase. In order to evaluate how well the algorithm is performing, a loss function needs to be identified, which is then minimized using the optimization algorithm during the training [16].

To train the network, it is necessary to choose and tune various hyperparameters, namely the loss and activation functions, the optimization algorithm, and the network topology.

Firstly, the loss function was identified as the mean squared error (MSE). This is the most suitable loss function for regression problems since it can be employed with almost all optimization algorithms [23]. The MSE is defined as

$$MSE = \sum_{i=1}^n \frac{(y_i - \hat{y}_i)^2}{n} \quad (2)$$

where \hat{y}_i is the output of the network and y_i is the actual value of the target.

For the activation functions in the output layer, a linear transfer function (*purelin*) was employed, whereas for the inner hidden layers, the hyperbolic tangent (*tansig*), the sigmoid (*logsig*), and the rectified linear (ReLU) transfer function (*poslin*) were identified as suitable since they can account for the nonlinearity of the data [22]. In order to understand which function is the most accurate for the problem that is being faced, their performances were evaluated and the analysis is reported in Section 4.

With regard to the optimization algorithm, three different options were considered, namely the Levenberg–Marquardt (LM), the Broyden–Fletcher–Goldfarb–Shanno (BFGS), and the Adam optimization algorithms [24]. As for the transfer function, a comparison of the results achieved with these algorithms is reported in the following section.

Finally, for the network topology, the numbers of neurons and hidden layers were changed and the ANN layout that minimizes the envelope-weighted mean absolute error (EMAE) was selected. The EMAE, which is a measure of the relative error of the model, is defined as [25]

$$EMAE = \frac{\sum_{i=1}^n |y_i - \hat{y}_i|}{\sum_{i=1}^n \max(y_i, \hat{y}_i)} * 100 \quad (3)$$

where y_i is the actual value of the target, \hat{y}_i is the predicted value and n is the total number of observations. Since this setting was problem dependent, the optimal ANN design was specified case by case. Initially, it was considered a single hidden layer network and the number of hidden layers was increased only if necessary. Moreover, for each target, in order to determine the optimal number of neurons per layer, a sensitivity analysis was carried out.

Once the hyperparameters were tuned and the model was trained, to assess its performance, four different parameters were evaluated:

- the envelope-weighted mean absolute error (EMAE), which is a measure of the relative error of the model and its expression is reported in Equation (3);
- the root mean square error (RMSE), which is a measure of the absolute error [19]:

$$RMSE = \sqrt{\sum_{i=1}^n \frac{(y_i - \hat{y}_i)^2}{n}} \quad (4)$$

where y_i is the actual value of the target, \hat{y}_i is the predicted value, and n is the total number of observations;

- the normalized root mean square error (nRMSE), whose expression was reported in Equation (1), which allows us to obtain a comparable measure of the absolute error of the model for the targets that have different scales [19,20];
- the coefficient of determination R^2 , which has values ranging from 0 to 1 and is a measure of how well the model is fitting the input data [19]:

$$R^2 = 1 - \frac{\sum_{i=1}^n (y_i - \hat{y}_i)^2}{\sum_{i=1}^n (y_i - \bar{y}_i)^2} \quad (5)$$

To build the ML model, feed-forward ANNs were adopted. These were implemented in MatLab by employing the functions *fitnet* and *train*. The first fits the ANN according to the number of hidden layers and the training algorithm specified, whereas the second trains the ANN once the inputs and targets have been identified. In addition, the data were split into training, testing, and validation datasets; 70% of the data were used for the training phase, 15% for the validation phase, and 15% for the model testing. The testing data were necessary for adjusting the values of the weights and biases of the network, while the validation dataset was used to stop the training of the model when the validation error started to increase so that the risk of overfitting was minimized. Finally, the testing data were used to test the ability of the generalization of the neural network by providing as inputs the samples that still had not been employed [16,17].

As defined previously, for the model developed, 3 targets were defined and 35 input variables were available. In order to include only the samples that referred to proper working conditions, each target was analyzed individually and sensor saturations or instantaneous solitary peaks were disregarded. In this way, 2.2% of the data were excluded from the analysis. In addition, the linear correlations between the different variables and, in particular, their relationships with the identified targets were highlighted. It emerged that the second target, namely the flow rate of steam going to the regenerator reboiler, was the most correlated. It had a very strong positive correlation not only with the variable representing the pressure of the same stream but also with the flow rates of the lean and rich amine entering and exiting the absorber. In addition, it also had a strong linear correlation with the temperature of the lean amine leaving the regenerator. Therefore, for the analysis of the steam rate, only the variables that had a linear correlation greater than 0.1 were included in the model. In this way, the number of inputs dropped to 26 and the computational cost was reduced. On the other hand, the other two targets did not show any significant linear correlations with the other variables. Thus, in these cases, all 35 inputs were included.

Finally, all the variables were normalized using a function called *mapminmax*, which makes all the values fall between -1 and 1 .

The methodology and workflow employed to develop the ML model are summarized in Figure 2. The figure shows the different steps that were carried out in the data analysis, sampling, and cleaning, after which it was possible to develop the ML model. This started with tuning the different hyperparameters, then, the actual model was implemented, and finally, the results were validated.

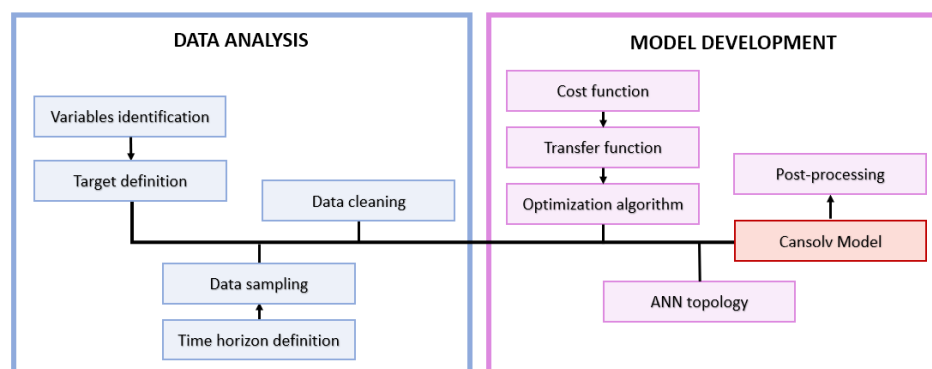


Figure 2. Scheme of the methodology employed to develop the Calsolv[®] model. Firstly, the data were analyzed, sampled, and cleaned. Then, after tuning the different hyperparameters, the model was developed and the results were validated.

4. Results

Three different models were developed, one for each target, in order to reduce the computational costs and increase the accuracy of the predictions.

In addition to the parameters and error values, the scatter plot of the target versus the output variables, where the data dispersion was highlighted, was reported for each of the analyzed cases. This plot allowed us to understand how well the dependent variable was predicted since it showed the distance between the actual value of the target and the corresponding output of the ANN. In addition, by also displaying the data dispersion, the number of samples that were well predicted can be easily understood.

4.1. SO₂ Flow Rate to the Claus Unit

The first analyzed target was the flow rate of SO₂ leaving the regenerator column and going to the Claus unit.

For this target, the effect of varying the activation function in the hidden layers and the optimization algorithm were studied. Table 2 shows a comparison of the results achieved on the testing data with three different activation functions, namely the sigmoid, hyperbolic tangent, and ReLU transfer functions.

Table 2. Hyperparameter tuning—comparison of the results for the prediction of the SO₂ flow rate with three different transfer functions: sigmoid, hyperbolic tangent, and ReLU.

	Sigmoid	Hyperbolic Tangent	ReLU
RMSE	2.97	2.78	2.95
R ²	0.808	0.824	0.797
Computational Time	39 min 15 s	37 min 50 s	27 min 51 s

What emerged from the simulations was that there was no significant difference in the computational time when the activation function in the hidden layers changed. However, it was easily noticeable that the hyperbolic transfer function ensured the best results both in terms of the RMSE and the R². Therefore, for the other targets, in all the ANNs implemented, the activation function employed in the hidden layers was chosen to be the hyperbolic tangent.

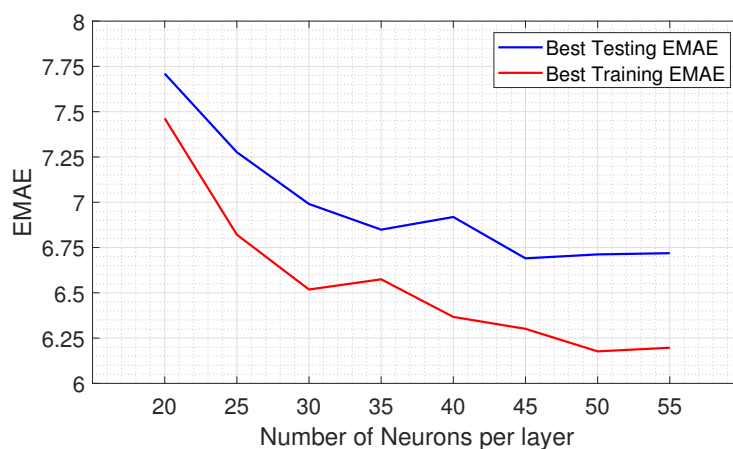
The results obtained by varying the optimization algorithm are reported in Table 3. Here, the performance achieved on the testing data is shown with regard to the LM, BFGS, and Adam optimization algorithms. The latter, however, required the tuning of the initial learning rate, whose optimal value was found to be equal to 0.0001 using a sensitivity analysis.

Table 3. Hyperparameter tuning—comparison of the results for the prediction of the SO₂ flow rate with three different optimization algorithms: Levenberg–Marquardt, BFGS quasi-Newton, and Adam.

	LM	BFGS	Adam
RMSE	2.78	3.37	2.96
R ²	0.824	0.680	0.741
Computational Time	37 min 50 s	1 h 48 min	4 h 19 min

The BFGS quasi-Newton method produced the worst results, whereas the LM algorithm proved to be the one that ensured the best performance of the model. Indeed, it showed the lowest value in terms of the RMSE and the highest value in terms of the coefficient of determination. In addition, it reached convergence faster than the other algorithms. These results are in accordance with the studies conducted by Hagan et al. and by Yu, which state that this algorithm ensures the fastest convergence when a regression problem is faced with neural networks that contain up to a few hundred weights [24,26]. Therefore, given both the advantages in terms of performance and computational time, the LM algorithm was used to train all the ANNs implemented in this study.

Finally, the topology of the ANN was optimized for each target. For an accurate prediction of the flow rate of SO₂, it was necessary to adopt a two-hidden-layer neural network and the best results were achieved with 45 neurons per layer. Figure 3 shows the sensitivity analysis that was carried out for this target in order to determine the optimal number of neurons. This was changed starting from 20 neurons per hidden layer up to 55 and the evolution of the EMAE was reported both on the training and testing data.

**Figure 3.** Evolution of the EMAE on the training (red line) and testing (blue line) datasets as the number of neurons in each hidden layer increased. Two identical layers were considered and the number of neurons ranged from 20 to 55 per layer.

What emerged from the analysis was that the error reduced up to 45 neurons in each layer. If the network complexity was increased further, the error did not decrease further but oscillated around a constant value both on the training and testing data. Therefore, for the target in question, a two-hidden layer ANN with 45 neurons per layer was selected.

The performance of the ML model for the prediction of the SO₂ flow rate is reported in Table 4 and Figure 4. The table shows the values of the errors evaluated on the training and testing data and the computational time required to complete one simulation. For all the models developed, and hence for all the targets analyzed, the same computational resources were employed, namely a computer with a processor with the following characteristics: Intel(R) Core(TM) i7-7820X CPU @ 3.60GHz 3.60 GHz. Figure 4 displays the regression plot of the output vs. target for the prediction, highlighting the data dispersion.

Table 4. Model performance for prediction of the SO₂ flow rate: values of the errors evaluated on the training and testing data and related computational time required.

	Test Performance	Train Performance
EMAE	6.79%	6.44%
RMSE	2.78	2.54
nRMSE	0.031	0.029
R ²	0.824	0.855
Computational Time	37 min 50 s	

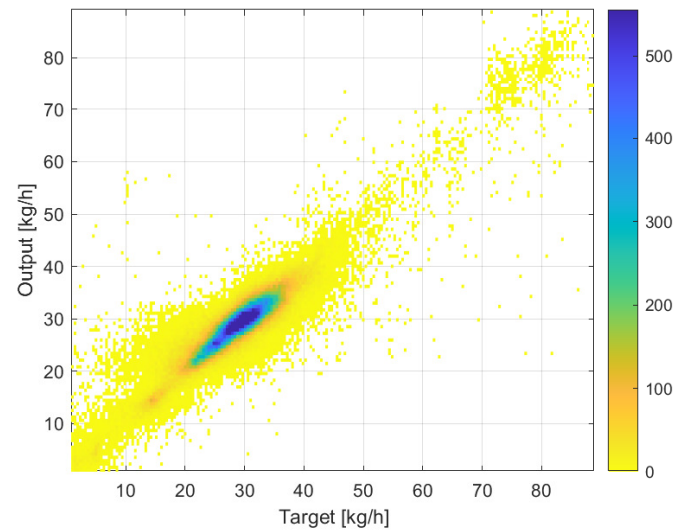


Figure 4. Regression plot of the output vs. the target for the prediction of the SO₂ flow rate that goes to the Claus unit, with the data dispersion highlighted.

Since the delta between the training and testing performance was not significant, as shown in Table 4, it can be concluded that there was no overfitting of the data and that the model had a good generalization capacity. In addition, the target was predicted with good accuracy. Indeed, the value of the coefficient of determination was greater than 0.8. The relative error was equal to 6.8% and the absolute error was around 2.5. This is a small value given that, as Figure 4 displays, the majority of the samples ranged between 30 and 40 kg/h.

The values reported in Table 4 are an indication of the overall performance of the model. In order to understand whether the prediction accuracy was good for the whole period considered and not just for some periods, the RMSE was evaluated every 20 days with a moving window. Its evolution is depicted in Figure 5.

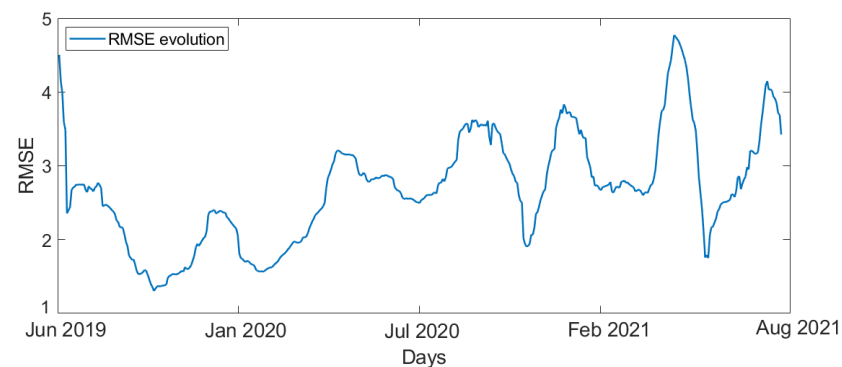


Figure 5. RMSE evolution calculated with a moving window over 20 days referring to the model's prediction of the SO₂ flow rate that goes to the Claus unit.

The RMSE mostly ranged between 2 and 3 and only during two limited time periods, namely June 2019 and March 2021, did it become higher than 4. During these months, the flow rate of SO₂ showed very nervous behaviour that the model developed could only predict with lower accuracy. However, even if there was a worsening of the performance for two months, the target mostly oscillated between 20 and 40 kg/h, which was a high value compared to the maximum value of the RMSE. Therefore, it can be concluded that the SO₂ flow rate was well predicted for the whole period considered.

4.2. Steam Flow Rate to the Regenerator

The second analyzed target was the flow rate of steam sent to the reboiler. To predict this target, a single-hidden-layer neural network was employed with 60 neurons in the hidden layer. In addition, given that the target was highly linearly correlated with various inputs, only the variables that had a correlation greater than 0.1 were included in the model. In this way, it was possible to disregard the less important features and include only 26 out of 35 input variables in the analysis.

The results obtained with the ANN model are reported in Table 5 and Figure 6.

Table 5. Model performance for prediction of the steam flow rate: values of the errors evaluated on the training and testing data and related computational time required.

	Test Performance	Train Performance
EMAE	0.663%	0.651%
RMSE	34.9	34.3
nRMSE	0.0084	0.0085
R ²	0.981	0.981
Computational Time	5 min 13 s	

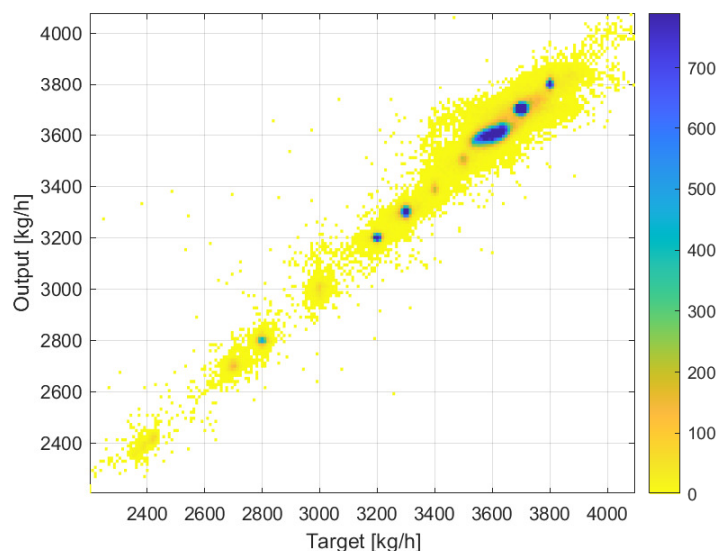


Figure 6. Regression plot of the output vs. the target for the prediction of the steam flow rate that goes to the regenerator, with the data dispersion highlighted.

The accuracy of the prediction was very high. Indeed, the value of the R² was higher than 0.95 and the nRMSE was four times lower than the previously analyzed target. Moreover, as shown in Figure 6, the great majority of the data fell on the diagonal. When analyzing the RMSE values, however, the scale of the target must be taken into consideration. Figure 6 shows that the target mostly oscillated between 3500 and 4000 kg/h. Therefore, even if in absolute terms, the value of 35 for the RMSE was high, in relative terms, it was not. Indeed, the EMAE, which is a measure of the relative error, was very low and close to 1%.

Moreover, the computational time was strongly reduced compared to the previously analyzed target. This was caused by both the adoption of a simpler ANN with only one hidden layer and the reduction of the input variables.

In addition, in this case, the RMSE value was evaluated with a moving window every 20 days in order to assess the model performance over the whole period analyzed. The RMSE evolution is reported in Figure 7.

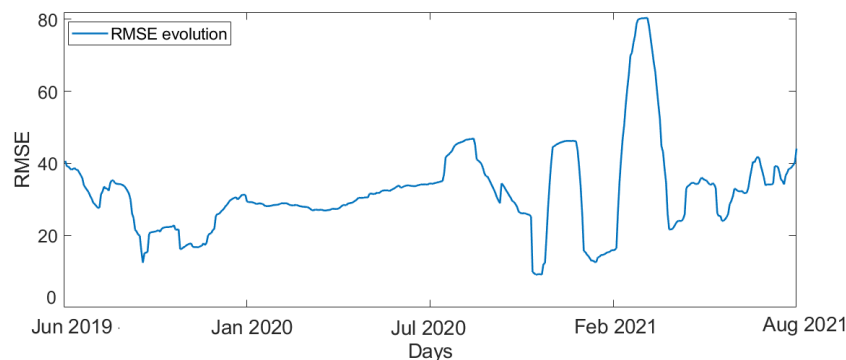


Figure 7. RMSE evolution calculated with a moving window over 20 days in terms of the model's prediction of the steam flow rate going to the regenerator.

The prediction was accurate for the whole time span considered. The RMSE was mostly lower than 40, except for February 2021, where it had a peak equal to 80. During this period, the target indeed had very high oscillations between 2000 and 4000 kg/h, which were more difficult to be reproduced by the model. However, even in this case given the scale of thousands of the target, this value indicated a relative error of about 1%, which was not a sign of the bad performance of the model.

4.3. SO₂ Concentration at the Absorber Outlet

The last target analyzed was the concentration of SO₂ in the fumes exiting from the top of the absorber and vented to the chimney. In this case, the best results were obtained with a 60-neuron single-hidden-layer ANN. The performance achieved is reported in Table 6 and Figure 8.

Table 6. Model performance for prediction of the SO₂ emissions: values of the errors evaluated on the training and testing data and related computational time required.

	Test Performance	Train Performance
EMAE	8.62%	8.03%
RMSE	2.06	1.69
RMSE	0.031	0.026
R ²	0.981	0.988
Computational Time	38 min 3 s	

The prediction accuracy was very high. Indeed, the value of the coefficient of determination was greater than 0.98 both on the training and testing data. In addition, the values of the errors, as reported in Table 6, were very close to the ones depicted in Table 4 in terms of the SO₂ flow rate. These targets had very similar scales, as seen in Figure 8, which shows that the majority of the samples of the SO₂ concentrations ranged between 0 and 40. Moreover, Figure 8 shows that there was no strong under- or overestimation of the data and that for almost all the samples, the actual value of the target and the value predicted by the ANN model coincided.

By looking at the R² values of the nRMSE and the EMAE, it is possible to conclude that the flow rate of steam was the target best described by the model. On the other hand,

the SO₂ flow rate that goes to the Claus unit was the target that the model predicted the worst. Even if the scale and the error values were very similar to those of the SO₂ concentration in the fumes, the R² value was significantly lower and there was a stronger over-/underestimation of the data.

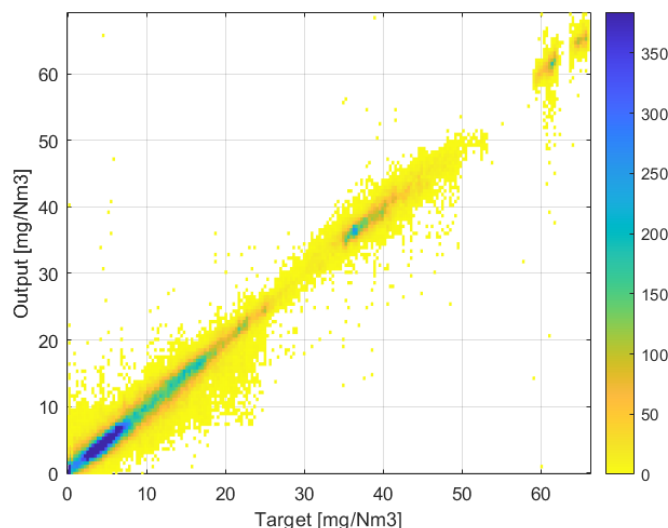


Figure 8. Regression plot of the output vs. the target for the prediction of the SO₂ concentration at the absorber outlet, with the data dispersion highlighted.

With regard to the computational time, in this case, more than 30 min was needed to train and test one ANN. This was very close to the time required for the first target analyzed. Therefore, the model built for the flow rate of steam for which about 5 min was necessary, was both the fastest to be trained and the one that produced the best results. This is because, as explained in Section 3, this target had a strong linear correlation with more than one variable.

Finally, for the concentration of SO₂ at the absorber outlet, the RMSE was analyzed for the whole period with a moving window over 20 days. As displayed in Figure 9, its magnitude was always below 2, except for May and July 2020, where two peaks with values higher than 3 were observed. During these two months, the target showed very stable behaviour and its value remained frozen, even for several days in a row. This evolution caused a worsening of the performance since the model developed was not able to accurately describe a very constant variable. However, even though during May and July 2020 the RMSE value increased, its magnitude was still low considering that the target during these months mostly oscillated between 15 and 40 mg/Nm³. Therefore, it is possible to conclude that the target could be accurately described for the whole period analyzed.

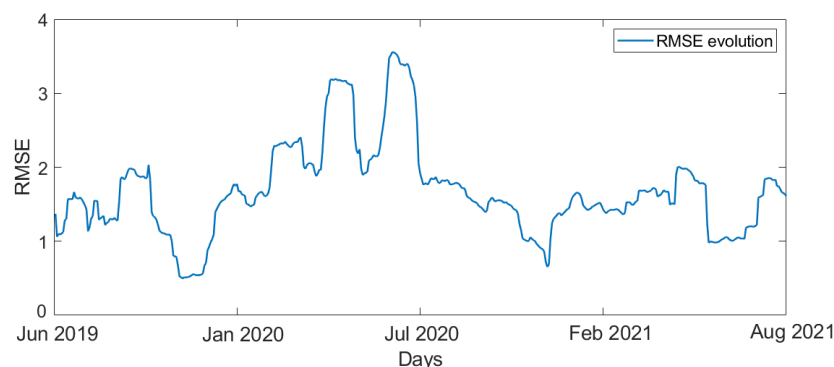


Figure 9. RMSE evolution calculated with a moving window over 20 days in terms of the model's prediction of the SO₂ emissions at the absorber outlet (concentration).

5. Conclusions

Given the results achieved and analyzed, it is possible to state that a model for the SO₂ removal system can be developed using purely data-driven techniques.

Three different models were implemented, one for each target, and all showed high accuracy in the results obtained. Particularly, for the second target analyzed, namely the flow rate of steam to the reboiler, excellent prediction accuracy was reached. Indeed, the value of the R² was always greater than 0.98 and the relative error was very low. The same conclusion can be drawn for the prediction of the SO₂ emissions for which a value of the R² parameter higher than 0.98 was achieved. In addition, the absolute error had very low values both on the training and testing datasets. These facts can also be observed in Figures 6 and 8, which show that nearly all the data were located on the diagonal. On the other hand, a slight under- and overestimation of the data occurred in the prediction of the SO₂ flow rate going to the Claus unit, as shown in Figure 4. However, even in this case, the R² was higher than 0.8 and the EMAE was about 6%, which are indications of good performance. Moreover, in all cases, there was no strong overfitting of the data and it can be said that the generalization capacity of the models built was excellent.

It can be concluded that ANNs are able to capture and describe the highly nonlinear relationships that occur between the variables of a chemical unit. In addition, the behaviour of a licensed process unit can be described properly using ML techniques and its outputs can be predicted with high accuracy. Therefore, with such a model, an optimal operating strategy for this unit can be defined. Moreover, this model completed the already existing physical simulation of the whole Eni plant. In this way, not only can the system be optimized but also the units placed upstream and downstream. This would result in an optimization of the entire plant, leading to an increase in its overall productivity by keeping the crucial parameters of the SO₂ removal technology under control and within the process limits.

A further development that would certainly complete this study would be to verify the models developed and the results obtained with the Eni system. In addition, it would be interesting to employ a dimensionality reduction technique such as principal component analysis (PCA). This would lead to a decrease in the computational time required to train one ANN since the number of input variables is shrunk. This has already been observed for the second target analyzed when the number of inputs was reduced from 35 to 26. Moreover, given the large number of variables of which the dataset is composed, a dimensionality reduction will also lead to the development of a more interpretable model [16].

Author Contributions: Conceptualization, F.G., M.M. and S.T. (Silvia Trimarchi); Data curation, M.M. and S.T. (Silvia Taddei); Formal analysis, F.G.; Investigation, S.T. (Silvia Taddei) and S.T. (Silvia Trimarchi); Methodology, F.G. and A.N.; Project administration, M.M.; Resources, M.M.; Software, S.T. (Silvia Taddei); Supervision, F.G. and S.T. (Silvia Taddei); Validation, A.N. and S.T. (Silvia Trimarchi); Visualization, A.N.; Writing—review and editing, A.N. and S.T. (Silvia Trimarchi). All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Restrictions apply to the availability of these data. Data was obtained from ENI SpA.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

AI	artificial intelligence
ANN	artificial neural network
APU	amine purification unit
BFGS	Broyden–Fletcher–Goldfarb–Shanno
EMAE	envelope-weighted mean absolute error
EOR	enhanced oil recovery

GPR	Gaussian process regression
LM	Levenberg–Marquardt
LSTM	long short-term memory
ML	machine learning
MSE	mean squared error
PCA	principal component analysis
nRMSE	normalized root mean square error
ReLU	rectified linear unit
RMSE	root mean square error
SVM	support vector machine

References

1. Benefits of Digitalisation to the Oil and Gas Industry. Available online: <https://futurecio.tech/benefits-of-digitalisation-to-the-oil-and-gas-industry/> (accessed on 10 November 2022).
2. Luca, C.; Gianmarco, R.; Lorenzo, L.; Danilo, L.; Andre, C.; Diletta, M.; Marco, M.; Elisabetta, P.; Piero, F.; Francesco, C.; et al. Digital Lighthouse: A Scalable Model for Digital Transformation in Oil & Gas. In Proceedings of the SPE EOR Conference at Oil and Gas West Asia, Adelaide, Australia, 17–19 October 2022.
3. Jafarinejad, S. Control and treatment of sulfur compounds specially sulfur oxides (SO_x) emissions from the petroleum industry: A review. *Chem. Int.* **2016**, *2*, 242–253.
4. World Health Organization. *WHO Global Air Quality Guidelines: Particulate Matter (PM_{2.5} and PM₁₀), Ozone, Nitrogen Dioxide, Sulfur Dioxide and Carbon Monoxide*; WHO: Geneva, Switzerland, 2021; p. 273.
5. Zhou, D.F.; Wu, Y.J.; Guo, J.B.; Lu, F.H. Analysis on the Main Problems of Industrial Application of the Regenerated Amine Desulphurization Technology. *Adv. Mater. Res.* **2014**, *881–883*, 42–47. [CrossRef]
6. Bangert, P. *Machine Learning and Data Science in the Oil and Gas Industry*; Gulf Professional Publishing: Oxford, UK, 2021.
7. Koroteev, D.; Tekic, Z. Artificial intelligence in oil and gas upstream: Trends, challenges, and scenarios for the future. *Energy AI* **2020**, *3*, 100041. [CrossRef]
8. Rauf, T.; Bakhshi, P.; Mirarab, M.; Shahbazi, K. Application of GA-Optimized ANNs to Predict the Water Content, CO₂ and H₂S Absorption Capacity of Diethanolamine (DEA) in Khangiran Gas Sweetening Plant. *Theor. Found. Chem. Eng.* **2020**, *54*, 995–1004. [CrossRef]
9. Koolivand, S.M.; Reza, A.; Hooman, A.; Hadis, K. Design of neural network for manipulating gas refinery sweetening regenerator column outputs. *Sep. Purif. Technol.* **2011**, *82*, 1–9. [CrossRef]
10. Kwon, H.; Oh, K.C.; Choi, Y.; Chung, Y.G.; Kim, J. Development and application of machine learning-based prediction model for distillation column. *Int. J. Intell. Syst.* **2020**, *36*, 1970–1997. [CrossRef]
11. Kundu, D.; Khanolkar, T.; Shah, T.; Bangad, S. Application of Machine Learning Technique to Predict Crude Distillation Column Inlet Temperature/Furnace Coil Outlet Temperature in Order to Maximize Distillate Yield and to Minimize Fuel Firing in Furnaces. *Int. J. Comput. Appl.* **2021**, *975*, 8887. [CrossRef]
12. Adib, H.; Sabet, A.; Naderifar, A.; Adib, M.; Ebrahimpzadeh, M. Evolving a prediction model based on machine learning approach for hydrogen sulfide removal from sour condensate of south pars natural gas processing plant. *J. Nat. Gas Sci. Eng.* **2015**, *27*, 74–81. [CrossRef]
13. Léveillé, V.; Claessens, T. Cansolv[®] SO₂ Scrubbing System: Review of commercial applications for smelter SO₂ emissions control. *J. S. Afr. Inst. Min. Metall.* **2009**, *109*, 485–489.
14. Towler, G.; Sinnott, R. *Chemical Engineering Design: Principles, Practice and Economics of Plant and Process Design*; Butterworth-Heinemann: Oxford, UK, 2021.
15. Preston, C. K.; Bruce, C.; Monea, M. J. An Update on the Integrated CCS Project at SaskPower’s Boundary Dam Power Station. In Proceedings of the 14th International Conference on Greenhouse Gas Control Technologies, Melbourne, Australia, 21–25 October 2018.
16. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; The MIT Press: Cambridge, MA, USA, 2016.
17. Géron, A. *Hands-on Machine Learning with Scikit-Learn, Keras and TensorFlow*; O’Reilly: Sebastopol, CA, USA, 2019.
18. Tariq, Z.; Aljawad, M.S.; Hasan, A.; Murtaza, M.; Mohammed, E.; El-Husseiny, A.; Sulaiman, A.A.; Mahmoud, M.; Abdulraheem, A. A systematic review of data science and machine learning applications to the oil and gas industry. *Pet. Res. Pet. Explor. Prod. Technol.* **2021**, *11*, 4339–4374. [CrossRef]
19. Chicchitelli, G.; D’Urso, P.; Minozzo, M. *Statistics: Principles and Methods*; Pearson: London, UK, 2021.
20. Vannitsem, S.; Daniel, S.W.; Messner, J. *Statistical Postprocessing of Ensemble Forecasts*; Elsevier: Amsterdam, The Netherlands, 2018.
21. Giuliani, M.; Cadei, L.; Montini, M.; Bianco, A.; Niccolai, A.; Mussetta, M.; Grimaccia, F. Hybrid Artificial Intelligence Techniques for Automatic Simulation Models Matching with Field Data. In Proceedings of the Abu Dhabi International Petroleum Exhibition and Conference, Abu Dhabi, United Arab Emirates, 12–15 November 2018.
22. Sharma, S.; Sharma, S.; Athaiya, A. Activation functions in neural networks. *Int. J. Eng. Appl. Sci. Technol.* **2020**, *4*, 310–316. [CrossRef]

23. Liu, Y.; Zou, C.; Chen, Q.; Zhao, J.; Wu, C. Optimization of Critical Parameters of Deep Learning for Electrical Resistivity Tomography to Identifying Hydrate. *Energies* **2022**, *15*, 4765. [[CrossRef](#)]
24. Hagan, M.T.; Demuth, H.B.; Beale, H.M.; De Jesús, O. *Neural Network Design*; Martin Hagan: Stillwater, OK, USA, 2014.
25. Niccolai, A.; Dolara, A.; Ogliari, E. Hybrid PV Power Forecasting Methods: A Comparison of Different Approaches. *Energies* **2021**, *14*, 451. [[CrossRef](#)]
26. Yu, H.; Wilamowski, B.M.W. Levenberg–Marquardt Training. In *Intelligent Systems*; CRC Press: Boca Raton, FL, USA, 2018.