An Ensemble Based Hybrid System for Residual Forecasting in Industrial Data

1st Luiz Felipe V. Verçosa
Dept. of Computer Engineering
University of Pernambuco
Recife, Brazil
lfvv@ecomp.poli.br

2nd Vildson da Rocha Borba
Dept. of Eletronic Engineering
University of Pernambuco
Recife, Brazil
vildson@gmail.com

3rd Paulo Hugo Espírito Santo Lima
Dept. of Telecommunications Engineering
University of Pernambuco
Recife, Brazil
paulo.hugo@poli.br

4th Luís Arturo Gómez Malagón
Dept. of Control and Automation Engineering
University of Pernambuco
Recife, Brazil
lagomezma@poli.br

5th Bruno Giublin
Ball Beverage Can South America S.A
Recife, Brazil
bruno.giublin@ball.com

6th João Fausto L. de Oliveira
Dept. of Computer Engineering
University of Pernambuco
Recife, Brazil
fausto.lorenzato@upe.br

***Abstract*—Time series forecasting has become an important task for several industrial processes. The employment of machine learning techniques also contributed to reduce costs and increase profits. In order to achieve accurate forecasts, it is important to perform a proper selection of parameters for the forecasting models. Hybrid sequential systems reduces model uncertainty through the employment of different models in the time series and residual series. In this work, an ensemble based hybrid sequential system is proposed, where a heterogeneous ensemble is used to perform residual forecasts. In this way, a pool of four models used in the industrial scenario are employed in order to achieve accurate forecasts. The experiments were conducted on real datasets from the industry in hourly averages and daily averages. The results show that the proposed system achieved promising results, and was able to improve the accuracy of the models in the pool in several cases.**

***Index Terms*—time series forecasting, hybrid systems, ensembles**

I. Introduction

Time series forecasting is an important task in many applications such as economy, sustainability, electricity, etc. The employment of techniques to increase industrial production and reduce costs have gained attention in the industry 4.0 scenario [1], [2]. In this sense, forecasting can be an important feature to analyze production, since it can be applied in soft sensors to estimate the next measured value, in predictive maintenance and in digital twins [3], [4]. An important goal of digital twins models is to predict the future sensors values and behavior of the system where an action is given. Therefore, forecasting models can be used for anomaly detection methods, maintenance and faults management, failures prediction, among other applications [5] [3], [4], [6]. Traditional forecasting models such as the autoregressive integrated moving average (ARIMA) have been employed in several industrial applications [7]–[9]. The popularity of ARIMA models is due to a well established procedure proposed by Box and Jenkins [10] to select a proper model, and the capacity to deal with nonstationary data. Nevertheless, one limitation of the ARIMA models is the assumption of a linear correlation structure in time series, which can reduce its performance in the presence of nonlinear patterns. Considering that real world data is often composed by a combination of linear and nonlinear patterns [11], the employment of the ARIMA model alone may not be sufficient to model the data.

Nonlinear models such as artificial neural networks (ANNs) and support vector machines (SVMs) have been widely employed in several time series forecasting applications, due to their ability to map nonlinear patterns. However, the employment of single nonlinear models may be challenging, since they can present problems such as model misspecification, overfitting and underfitting [12].

In order to overcome the limitations of single models (linear or nonlinear), hybrid systems have been proposed to improve the forecasting accuracy. In the literature, two hybrid systems have been explored with promising results: Sequential hybrid methods and ensemble methods. sequential hybrid methods assume that a time series can be decomposed in its linear and nonlinear components. Thus, such systems are often based on three steps: time series forecasting, residual forecasting and combination of forecasts [11], [13], [14]. In the first step, several works employ a linear model such as the ARIMA to perform time series forecasting. The error series (residuals) are calculated as the difference between time series and forecasts, and are used as an input data to the second step. On the second step, nonlinear models such as ANNs and SVMs are often employed. Since linear models can perform linear mappings in the series, it is assumed that the residuals are mostly comprised by nonlinear patterns which are mapped by a nonlinear model. The final step performs the combinations of the forecasts from the time series and residual series. Sequential hybrid systems have been employed in several applications such as electricity [15], environmental sciences [16], etc.

Sequential hybrid models have achieved promising results in the literature, however despite its assumption of decomposition of the time series in its linear and nonlinear counterparts, some problems remain unsolved. For instance, the residual data may present random fluctuations and heteroscedastic patterns, which can pose a challenge for the specification of nonlinear models [17], [18].

Ensemble methods can be employed to reduce the risk of selection of a misspecified model through the combination of several model. In order to ensure the improvement in accuracy, the models in the pool must be accurate and diverse from each other. The errors obtained in the forecasting process can be decomposed in errors caused by difference between the average model and the time series (bias error) and the difference between the single model and the average model (variance error). Therefore, the bias-variance decomposition of the error is presented in eq 1, where the target value and past data of the time series are represented by τ and $\mathbf{z}$ respectively, and E is an expectation operator.

$$E[\tau - f(\mathbf{z})]^2 = \{\tau - E[f(\mathbf{z})]\}^2 + E(\{E[f(\mathbf{z})] - f(\mathbf{z})\}^2). \quad (1)$$

Ensemble methods can reduce model uncertainty by reducing the variance error without increasing the bias [19], [20], since several models are considered in the combination. However, in order improve the accuracy of the system, the models must be diverse. The diversity may be achieved through the employment of distinct models in the pool (heterogeneous ensemble), while using the same models with different parameters are referred to as homogeneous ensembles.

In this work an ensemble based hybrid sequential system is proposed to improve the accuracy of forecasts of the system through the employment of a heterogeneous ensemble to perform residual forecasts. The combination of forecasts considering the models in the pool is explored using four different operators. Moreover, the novelty of the proposed method relies in the architecture of the proposed system composed of different forecasting methods and combination operators for residual forecasting. In this sense, the proposed method present the following advantages over the methods of the literature:

- The employment of a heterogeneous ensemble to forecast residuals.
- Versatility in the use of the combination operators of the ensemble.
- Capability to improve the forecasting accuracy by reducing the variance error.

The remainder of this paper is structured as follows: Section II presents the related works in the literature of industrial forecasting and hybrid sequential systems. The proposed method is explained in Section III, the experiments and discussion are described in Section IV and the conclusion is given in Section V.

II. Related Works

In general, the literature of time series presents several forecasting approaches such as single, ensemble and hybrid models.

The employment of single models consists on obtaining forecasting values from a statistical or machine learning based model. In this sense, statistical linear models such as the ARIMA model have been used in nonstationary data in industrial applications [21], [22] in order to improve efficiency. In the light of the limitations of the abovementioned model, nonlinear models such as ANNs, SVMs [23], [24] have also been considered to perform forecasts. However, traditional machine learning methods such as ANNs may present less number of processing layers (shallow architectures), which can reduce generalization capacity of the model. In contrast, deep neural networks (DNNs) are able to deal with complex nonlinear patterns in large data by using higher number of processing layers [25]. One example of DNN is the long short-term memory (LSTM) through which is possible to achieve promising results in the industrial area [25].

Modelling industrial processes often pose a challenge in the selection of an appropriate model to perform forecasts. Considering the large amount of available data and its complex nonlinear characteristics, the combination of models through the employment of ensemble techniques have achieved promising results [26]–[28]. Soares *et al.* [26] used bootstrap and noise injection approaches to ensure diversity in the ensemble in soft sensor design with small datasets. Chen *et al.* [27] proposed an adaptive ensemble method to combine local models, using the Euclidean distance as a similarity measure. Liu *et al.* [28] proposed the combination of global and local Gaussian process regression (GPR) models to forecast industrial processes variables. Another promising deep learning approach in the design of soft sensors is the eXtreme Gradient Boosting (XGBoost) [29]. XGBoost is based on decision-tree ensemble algorithm, optimized by an improved gradient boosting [30] and has been applied to diverse prediction tasks [31].

Hybrid systems are often designed to improve the accuracy of single models in different tasks. In the time series forecasting scenario, the use of sequential hybrid systems have gained attention since it can achieve better results than that of single and ensemble models [17], [20]. Sequential hybrid systems are mainly composed of three stages:

- Time series forecasting
- Residual series forecasting
- Combination of forecasts.

Such systems differ from ensembles in several aspects. The first is that, the models are trained on different data sets, which are the time series and the residual series. The residual series is obtained from the difference between actual values from the series and forecasts achieved by some forecasting model. The second aspect is that a second model is used to forecast residual values. And lastly, the combination is not a fusion of forecasts, and different combination operators are employed. For instance, Zhang [11] employed an ARIMA model to perform a linear time series forecasting ($\hat{L}_t$), a Multilayer Perceptron (MLP) in nonlinear residual forecasting $\hat{N}_t$ and assumed a linear relationship between linear and nonlinear forecasts through the employment of a summation operator, thus the final forecast achieved is presented in eq. 2

$$F_t = \hat{L}_t + \hat{N}_t. \quad (2)$$

For the sake of simplicity, the representation of this architecture will be henceforward referred to as ARIMA+MLP, representing the ARIMA model in the time series, the MLP used in the residual series, and the summation operator for the combination of forecasts. The same architecture was used in several works in the literature, using different approaches. Panigrahi and Behera [13] used an exponential smoothing (ETS) model to perform linear forecasts and a MLP in the residuals, followed by a summation of forecasts, resulting in an ETS+MLP hybrid model. Pai and Lin [14] employed an ARIMA+SVR to forecast stock prices. Fan *et al.* [32] employed an ARIMA+LSTM hybrid system in forecasting oil and gas production.

However, the relation between the linear ($\hat{L}_t$) and nonlinear ($\hat{N}_t$) forecasts in the combination stage may not always be linear. In this sense, de Mattos Neto *et al.* [16] employed a nonlinear function to perform the combination of forecasts, and the final forecast achieved is presented in eq. 3

$$\hat{F}_t = f(\hat{L}_t, \hat{N}_t), \quad (3)$$

where f is the nonlinear function used in the combination. In this work, the MLP and SVR are employed as combination functions. A similar concept was used in the work of Khashei and Bijari [33]. Domingos *et al.* [34] presented and improved architecture, where past forecast values from linear and nonlinear models are used as input for the combination function.

The residual forecasting stage is challenging since the residual series may present irregular patterns and heteroscedasticity. In order to address this problem in the residual forecasting stage, Holanda and Oliveira [18] employed a swarm based ensemble (SBE) to perform forecasts in residual residual series. The method, named ARIMA+SBE, used a particle swarm optimization (PSO) [35] algorithm in order to optimize a SVR mode. The residual forecast was given by the best optimized particle and the average/median forecast from the swarm and using part of the population.

However, the employment of swarm based methods to perform the generation of the forecasting methods may converge to similar locations in the optimization space, producing less diverse forecasting models. In this way, the proposed method addresses this problem through the employment of a heterogeneous ensemble composed by forecasting methods used in the Industrial scenario (MLP, SVR, XGBoost, LSTM), and the combinations are based on the average/median of the base models.

III. Proposed Method

The proposed hybrid system is comprised of three stages: time series forecasting, residual forecasting and combination of forecasts. However, there are two combinations in the proposed hybrid system: one is in the residual forecasting stage, where the forecasts of individual models from a heterogeneous ensemble are combined through the use of distinct operators, and the last combines linear and nonlinear forecasts.

A. Time series forecasting

Fig. 1 presents the architecture of the proposed system. The first stage consists of performing linear modeling of the time series through the employment of an ARIMA model, defined in eq. 4.

$$\begin{aligned}\mathbf{Z}_t = \mu + \phi_1 z_{t-1} + \phi_2 z_{t-2} + ... + \phi_p z_{t-p} + \epsilon_t \\ -\theta_1 \epsilon_{t-1} - \theta_2 \epsilon_{t-2} - ... - \theta_q \epsilon_{t-q},\end{aligned} \quad (4)$$

where z_t and ϵ_t are the time series value and random error at time t, respectively; μ, $\phi_1, \phi_2, ..., \phi_p$ and $\theta_1, \theta_2, ..., \theta_q$ are the intercept, the autoregressive coefficients and the moving average coefficients, respectively. The p and q are integers and refer to the model orders. The ARIMA model is estimated using the time series (Z_t) as input, and then linear forecasts ($\hat{L}_t$) are produced. The residual series (E_t) is calculated by the difference between the time series and linear forecasts as shown in eq. 5

$$E_t = Z_t - \hat{L}_t. \quad (5)$$

The residual series is used as input in the second stage, of the proposed system.

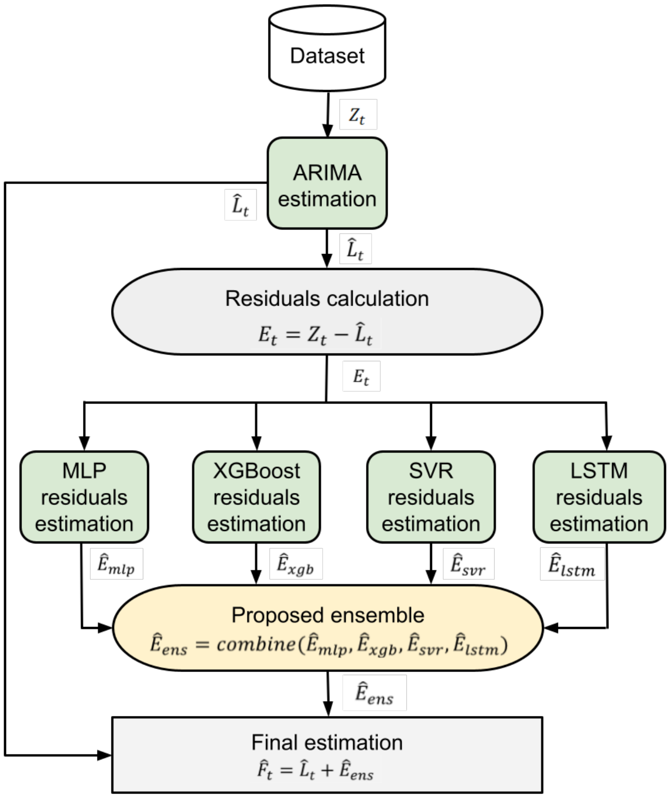

Fig. 1. Architecture of proposed hybrid system.

B. Residual series forecasting

In the second stage, nonlinear forecasts are obtained from a pool of singular models composed by the MLP [11], XGBoost [30], SVR [36] and LSTM [37], as presented in equation 6, where g is a nonlinear function which considers as input past residual values (lags) from E_t, where e_{t-i} represent the residual value at lag i.

$$\hat{E}_t = g(e_{t-1}, e_{t-2}, ..., e_{t-l}). \tag{6}$$

Each model in the pool is trained independently using the residual series. Once the training step is completed, the residual forecasts of each model can be obtained, being $\hat{E}_{mlp}, \hat{E}_{xgb}, \hat{E}_{svr}, \hat{E}_{lstm}$ the forecasts from the MLP, XGBoost, SVR and LSTM, respectively.

In order to achieve the final residual forecast (E_{ens}), the forecasts obtained by each model in the pool are combined as shown in eq. 7, where $combine$ is an operator used to combine the singular residual forecasts.

$$\hat{E}_{ens} = combine(\hat{E}_{mlp}, \hat{E}_{xgb}, \hat{E}_{svr}, \hat{E}_{lstm}). \tag{7}$$

In this stage classic ensemble combination operators for time series forecasting could be employed. In this work, the following combination strategies are considered:

- Mean of the forecasts
- Median of the forecasts
- Mean of the best x models (Mean$_x$)
- Ordinary least squares (OLS)

The median operator is often preferred when combining forecasts [38], [39] since it is less sensible to outliers when compared to the mean operator. The Mean$_x$ operator employs the mean operator in the best x forecasting models, where x is the number of models, based on the error from the validation set. The OLS was also considered for combining the forecasts. Let $O_{n \times x}$ be a matrix of forecasts for a given dataset, where n is the number of forecasts and x the number of models, therefore each column in the matrix represents the forecasts of one model. The combination of forecasts using OLS consists of finding weight matrix β as presented in eq. 8.

$$\beta = O^{-1} \times Z_t \tag{8}$$

In this way, the residual combination of forecasts will be given by a weighted sum of individual forecasts.

C. Final forecast

The last step in the proposed model consists on the final combinations of the forecasts achieved in previous steps. The final forecast ($\hat{F}_t$) is achieved by a summation of linear $\hat{L}_t$ and nonlinear $\hat{E}_{ens}$ as presented in eq. 9.

$$\hat{F}_t = \hat{L}_t + \hat{E}_{ens} \tag{9}$$

The summation operator was considered in this step to analyse the contribution of the heterogeneous models in the previous steps.

IV. Experiments and Results

The experiments were conducted in real datasets from the industry, regarding the operating sensors values of production machines. Specifically, the environment is composed of a Bodymaker machine in order to produce cans. The data was collected every 15 minutes from august 20th 2020 to february 4th 2021, resulting in approximately 168 days of records. The data represent pressure and temperature measurements of different sensors: oil pressure from a spindle motor, coolant temperature, water temperature, oil temperature and redraw pressure. The abbreviations for the data series names are presented in Table I. The monitoring of these data is important to evaluate the production process.

TABLE I
Dataset series with respective abbreviation

Name	Abbreviation
Spindle Oil Pressure	Oil_Pressure
Coolant Temperature	Coolant_Temp
Water temperature	Water_Temp
Oil Temperature	Oil_Temp
Redraw pressure	Redraw_Pressure

The proposed system was compared with methods used in the forecast of industrial data, such as ARIMA, SVM, MLP, LSTM and XGBoost. Furthermore, hybrid systems in the literature such as ARIMA+MLP [11], ARIMA+SVR [14], ARIMA+LSTM [32] were also used in the experiments. To the best of the authors' knowledge, the ARIMA+XGBOOST was not explored in previous studies.

In order to better evaluate the performance of the models, the datasets were split in three subsets: training (60% of the data), validation (20%) and testing (20%). In order to select the best set of parameters for each dataset, a grid search procedure was conducted, where the best configuration with lowest MSE in the validation set was selected. The parameters used in the grid search are presented in Table II. It is important to mention that the same parameters presented in Table II are used in the search process in the hybrid systems and single models.

Furthermore, the data sets were scaled to the [0..1] interval according to Eq. 10, where Z_{norm} is the scaled series, Z_{max} and Z_{min} are the maximum and minimum values of the series, respectively.

$$Z_{norm} = \frac{Z - Z_{min}}{Z_{max} - Z_{min}} \tag{10}$$

Simulations were performed in two scenarios: the first considering hourly averages and the second one, daily averages. The accuracy of the models is given by the mean squared error (MSE) and the mean absolute percentage error (MAPE) presented in Eqs. 11 and 12, respectively.

$$MSE = \frac{1}{n}\sum_{t=1}^{n}(Z_t - \hat{F}_t)^2 \tag{11}$$

TABLE II
VALUES FOR THE PARAMETERS OF THE MODELS

Model	**Parameters**	**Values**
ARIMA	p, d, q	Hyndman [40] methodology
MLP	Number of neurons	[10, 50, 100, 200, 400]
	Learing Rate	[0.0001, 0.001, 0.01, 0.1]
	Early Stop	[’True’, ’False’]
	Epochs	[200, 400, 800]
	Algorithm	[“Adam”, ’SGD’]
	lag	[1,...,100]
XGBoost	Learning Rate	[0.001, 0.01, 0.2]
	Booster	[“gbtree”,’gblinear’]
	Max depth	[3, 4, 5, 8]
	Number of estimators	[5, 10, 15, 25, 50, 100]
	Minimum Child Weight	[1, 2, 3]
	Subsample	[0.8, 0.9, 1]
	Colsamble bytree	[0.7, 0.8, 0.9, 1]
	lag	[1,...,100]
SVR	C	[10^{-1}, 10^{0}, 10^{1},10^{2}]
	gamma	[10^{-2} ,10^{-1}, 10^{0}, 10^{1},10^{2}]
	epsilon	[10^{-3} ,10^{-2}, 10^{-1}]
	Kernel	RBF
	lag	[1,...,100]
LSTM	Number of Neurons	[4, 16, 64, 200]
	Epochs	[25, 100, 200, 400]
	Lags	[1,...,100]

$$MAPE = 100 \times \frac{1}{n} \sum_{t=1}^{n} \left| \frac{Z_t - \hat{F}_t}{Z_t} \right| \qquad (12)$$

The MSE penalizes higher errors, however it is sensitive to outliers and is scale dependent [41], thus comparisons between different datasets should be avoided when considering this metric. In contrast, MAPE produces an absolute percentage error which is not scale dependent, nevertheless it is an asymmetric measure, which can measure differently positive and negative errors [42].

Furthermore, a percentage of improvement (PI) metric over the ARIMA model is employed to analyze the improvements of the alternative models. The PI is defined in Eq. 13, where MSE_{ARIMA} and MSE_{Model} are the MSEs achieved by the ARIMA and the compared model respectively.

$$PI = 100 \times (\frac{MSE_{ARIMA} - MSE_{Model}}{MSE_{ARIMA}}) \qquad (13)$$

Positive values of PI indicate an increase of performance of the analyzed model over the ARIMA regarding the considered metric, otherwise negative values are produced. The same equation can be employed for other metrics such as MAPE.

For reasons of simplicity, the proposed method will be represented by the approach used in the residual combination stage. Thus, in the simulations, each approach will be referenced as described in Table III.

TABLE III
NOMENCLATURE OF THE PROPOSED METHOD WITH THE COMBINATION APPROACHES

Abbreviation	Detail
$\text{Proposed}_{\text{Mean}}$	Mean of forecasts
$\text{Proposed}_{\text{Median}}$	Median of forecasts
$\text{Proposed}_{\text{Mean}_\text{x}}$	Mean of best x models
$\text{Proposed}_{\text{OLS}_{(\text{Mdl}_1,\text{Mdl}_2)}}$	OLS using Models Mdl_1 and Mdl_2

A. Simulations with hourly averages

In this simulation, hourly averages from the original data were considered, resulting in approximately 4043 records for each dataset. Therefore, the amount of data used in the training, validation and testing sets were 2425, 809 and 809 respectively.

The best results obtained in the simulations are presented in Table IV. The proposed methods achieved the best results in datasets Coolant_Temp, Water_Temp and Oil_Pressure for the MSE metric. Furthermore, the best results were obtained by the Mean, Mean_2 and Median operators. The ARIMA+MLP achieved the lowest MSE for dataset Oil_Temp. The ARIMA+XGB obtained lowest MSE for Redraw_Pressure.

When considering the MAPE metric, the proposed methods obtained the best results in Oil_Pressure. ARIMA+MLP outperformed the other methods in datasets Coolant_Temp and Oil_Temp. SVR single obtained the lowest error for Water_Temp and ARIMA+SVR was the best in Redraw_Pressure dataset.

The Percentage Improvement (PI) is presented on Table V. The Mean and Median based operators could improve the results over the ARIMA model in most datasets regarding the MSE and MAPE metrics. Only in few cases they worsened ARIMA results. In contrast, the OLS operator only improved the ARIMA model in few cases and worsened its performance in a reasonable number of cases.

B. Simulations with daily data

Using daily averages, the total of records is approximately 168 from which 100 records were used for training, 34 for the validation set and 34 in the test set.

The results of the simulations are presented in Table VI. The proposed method achieved the best MSE in the Oil_Pressure, Water_Temp and Oil_Temp datasets with the Mean and Median based operators. The ARIMA+LSTM achieved the best MSE in the Oil_Pressure dataset.The MLP single achieved the best MSE in the Coolant_Temp dataset.

Regarding the performance using the MAPE metric, the SVR single model achieved the best results in datasets Coolant_Temp, Water_Temp and Oil_Temp. The LSTM single, achieved the best results in the Oil_Pressure dataset. The ARIMA single, achieved the best results in the Redraw_Pressure dataset.

TABLE IV

MSE AND MAPE VALUES FOR SINGLE, HYBRID AND ENSEMBLE MODELS, FOR THE HOURLY AVERAGE SCENARIO

Model	**Oil_Pressure**		**Coolant_Temp**		**Water_Temp**		**Oil_Temp**		**Redraw_Pressure**	
	MAPE	MSE	MAPE	MSE	MAPE	MSE	MAPE	MSE	MAPE	MSE
ARIMA	3.4625	3.2961	1.1037	65.4155	2.6611	115.9752	4.3148	647.0621	3.3959	15.2706
LSTM	3.5297	3.4260	1.3035	79.3522	2.7987	129.6995	4.1428	635.1076	3.4990	17.8932
SVR	3.4149	3.2537	1.3709	90.4133	**2.4531**	125.1537	4.2126	717.5034	3.8653	21.0488
MLP	3.4642	3.2516	1.4917	112.6258	2.8171	131.3245	4.2131	617.6399	3.4007	15.6046
XGBOOST	3.5728	3.4660	1.5971	113.1894	2.6368	127.6695	4.3531	677.7086	3.8494	20.3075
ARIMA + MLP	3.3705	3.1537	**1.1001**	65.4543	2.6934	115.2826	**4.1265**	**586.2100**	3.3974	15.4860
ARIMA + XGB	3.4134	3.1600	1.1056	65.6992	2.6950	122.2328	4.2708	630.8521	3.3912	**15.1646**
ARIMA + SVR	3.3999	3.1750	1.1185	66.4728	2.6979	117.8294	4.3640	659.0274	**3.3202**	15.5141
ARIMA + LSTM	3.3951	3.1766	1.1978	76.3058	2.5479	114.4917	4.2658	621.8902	3.3819	15.2320
$\text{Proposed}_{\text{Mean}}$	3.3811	**3.1283**	1.1011	65.4396	2.6536	113.3591	4.2428	618.0846	3.3560	15.2564
$\text{Proposed}_{\text{Mean}_2}$	**3.3681**	3.1361	1.1010	65.4903	2.5953	**111.8834**	4.1913	599.8784	3.3437	15.3010
$\text{Proposed}_{\text{Median}}$	3.3902	3.1449	1.1029	**65.3615**	2.6045	112.4022	4.2473	618.3024	3.3654	15.2248
$\text{Proposed}_{\text{OLS}_{(\text{MLP,SVR})}}$	3.3750	3.1551	1.1707	68.8959	2.6937	118.0807	4.3442	651.1755	3.3493	15.9573
$\text{Proposed}_{\text{OLS}_{(\text{MLP,XGBoost})}}$	3.4253	3.2260	1.1088	65.5081	2.6520	116.6744	4.2927	638.5621	3.4184	15.4120

The PI over the ARIMA is presented in Table VII. In general, the proposed method improved the performance of the ARIMA in several datasets regarding MSE and MAPE. Nevertheless, the single models LSTM, MLP and SVR also achieved the best improvements in datasets Oil_Pressure, Coolant_Temp, Water_Temp and Oil_Temp.

C. Discussion

In general, the proposed hybrid system achieved the best results in the majority of datasets considering the daily and hourly averages scenarios. The mean and median based operators achieved the overall best results. It is important to notice that the best method based on the MSE was not necessarily the one with best MAPE, this is due to the asymmetry of the MAPE metric which may evaluate differently positive and negative forecasting errors [41].

The hybrid systems such as ARIMA+MLP and ARIMA+SVR also improved the results over the ARIMA technique, and outperformed the single models in several cases. Negative results in Tables V and VII indicate that there was not a improvement over the ARIMA dataset. Methods that employ stochastic gradient based learning algorithms such as MLP often suffer from local minimum problems during the training process. On the other hand, the SVR employs a quadratic optimization procedure which has a single minimum. A drawback from SVR is that it is very sensible to the hyperparameters. Therefore, the combination from different techniques in the residual forecasting phase could improve the final forecasting accuracy. This result reinforce the hypothesis that employing ensemble methods in the residual forecasting stage can improve the accuracy of the system.

V. Conclusion

In this work a heterogeneous ensemble was proposed to improve the performance in the residual forecasting stage in the context of sequential hybrid systems using industrial data. The residual forecasting stage is challenging for single models, since it may present irregular patterns and heteroscedastic characteristics.

The proposed model employed forecasting models used in the literature to compose the pool of forecasts, promoting the diversity among forecasts. The results obtained in the experiment were given by the MSE and MAPE metrics, and shows that even though the proposed method did not achieve the best results in all datasets, it could improve the results of the members in the pool.

The proposed work employs a grid search method to search for the best set of parameters on the validation set. This could be improved through the use of meta-heuristic search algorithms such as particle swarm optimization. Moreover, it is important to mention that the architecture of the hybrid system can be explored in several ways, allowing the usage of different combination operators, members in the pool or trying different models in the first stage.

Thus, for future works, an automatic ensemble selection algorithm will be employed to select the best models in the pool in order to improve the accuracy of the system. Another possibility is the employment of nonlinear functions in the final forecast stage.

Acknowledgement

The authors' would like to thank Ball Beverage Can South America S.A, for providing the data and support for this research. This work was supported by the Coordination for the Improvement of Higher Education Personnel - Brazil (CAPES)- Financing Code 001.

References

[1] L. Patanè and M. G. Xibilia, "Echo-state networks for soft sensor design in an sru process," *Information Sciences*, vol. 566, pp. 195–214, 2021.

TABLE V
PERCENTAGE IMPROVEMENT OVER THE ARIMA FOR THE HOURLY AVERAGE SCENARIO

Model	**Oil_Pressure**		**Coolant_temp**		**Water_temp**		**Oil_temp**		**Redraw_pressure**	
	MAPE	MSE	MAPE	MSE	MAPE	MSE	MAPE	MSE	MAPE	MSE
LSTM	-1.94	-3.94	-18.1	-21.3	-5.17	-11.83	3.99	1.85	-3.04	-17.17
SVR	1.37	1.29	-24.21	-38.21	**7.82**	-7.91	2.37	-10.89	-13.82	-37.84
MLP	-0.05	1.35	-35.15	-72.17	-5.86	-13.23	2.36	4.55	-0.14	-2.19
XGBOOST	-3.19	-5.15	-44.7	-73.03	0.91	-10.08	- 0.89	-4.74	-13.35	-32.98
ARIMA+MLP	2.66	4.32	**0.33**	-0.06	-1.21	0.6	**4.36**	**9.4**	-0.04	-1.41
ARIMA+XGB	1.42	4.13	- 0.17	-0.43	-1.27	-5.4	1.02	2.51	0.14	**0.69**
ARIMA+SVR	1.81	3.67	-1.34	-1.62	-1.38	-1.6	-1.14	-1.85	**2.23**	-1.59
ARIMA+LSTM	1.95	3.63	-8.53	-16.65	4.25	1.28	1.14	3.89	.41	0.25
Proposed$_{\mathrm{Mean}}$	2.35	**5.09**	0.24	-0.04	0.28	2.26	1.67	4.48	1.17	0.09
Proposed$_{\mathrm{Mean_2}}$	**2.72**	4.85	0.24	-0.11	2.47	**3.53**	2.86	7.29	1.54	-0.20
Proposed$_{\mathrm{Median}}$	2.09	4.59	0.07	**0.08**	2.13	3.08	1.56	4.44	0.9	0.3
Proposed$_{\mathrm{OLS_{(MLP,SVR)}}}$	2.53	4.28	-6.07	-5.32	-1.23	-1.82	-0.68	-0.64	1.37	-4.5
Proposed$_{\mathrm{OLS_{(MLP,XGBoost)}}}$	1.07	2.13	- 0.46	- 0.14	0.34	-0.6	0.51	1.31	- .66	-0.93

TABLE VI
MSE AND MAPE VALUES FOR SINGLE, HYBRID AND ENSEMBLE MODELS, FOR DAILY AVERAGE SCENARIO.

Model	**Oil_Pressure**		**Coolant_Temp**		**Water_Temp**		**Oil_Temp**		**Redraw_Pressure**	
	MAPE	MSE	MAPE	MSE	MAPE	MSE	MAPE	MSE	MAPE	MSE
ARIMA	1.5980	0.7484	2.0346	284.8309	1.4360	44.8832	1.6265	151.6869	**1.6710**	4.0097
LSTM	**1.1417**	0.4465	1.9701	185.0612	1.3477	43.7084	1.8719	183.4404	1.8110	4.3893
SVR	1.5556	0.7314	**1.7012**	167.1959	**1.3415**	45.2243	**1.5443**	153.8141	2.5376	7.8484
MLP	1.4499	0.6713	1.9210	**159.7138**	1.3451	44.8016	3.4850	445.1252	2.2089	7.0168
XGBOOST	1.5842	0.7300	1.8159	200.1074	1.3758	45.7443	1.6949	159.4432	2.4962	7.6281
ARIMA + MLP	1.5470	0.6576	2.9023	406.8152	1.5729	48.5009	2.0976	182.2708	1.6946	4.5731
ARIMA + XGB	1.2451	0.4544	1.8159	276.0913	1.5983	48.2902	1.5972	151.2316	1.7280	4.2032
ARIMA + SVR	2.0435	1.2542	2.0873	224.7340	1.5519	45.2171	1.6110	151.4154	1.6800	4.0552
ARIMA + LSTM	1.2537	0.4300	2.0200	294.8302	1.5273	46.8749	1.6302	151.6100	1.6665	**3.9684**
Proposed$_{\mathrm{Mean}}$	1.4316	0.5633	2.0060	271.9830	1.3947	**42.3018**	1.6786	150.7060	1.6794	4.0653
Proposed$_{\mathrm{Mean_2}}$	1.2343	**0.4221**	1.9728	232.5375	1.4097	43.4680	1.6016	151.2204	1.6720	4.0063
Proposed$_{\mathrm{Median}}$	1.3538	0.5062	1.9933	282.4820	1.4369	44.4583	1.6107	**150.6423**	1.6852	4.0591
Proposed$_{\mathrm{OLS_{(MLP,SVR)}}}$	1.4806	0.6715	2.0308	222.0162	1.6827	51.4489	1.6252	153.8845	1.6842	4.0515
Proposed$_{\mathrm{OLS_{(MLP,XGBoost)}}}$	1.9339	1.1103	2.0489	223.4944	1.4160	43.9589	1.7207	156.5294	1.7217	4.2288

[2] L. Fortuna, S. Graziani, A. Rizzo, M. G. Xibilia *et al.*, *Soft sensors for monitoring and control of industrial processes*. Springer, 2007, vol. 22.

[3] W. Luo, T. Hu, C. Zhang, and Y. Wei, "Digital twin for cnc machine tool: modeling and using strategy," *Journal of Ambient Intelligence and Humanized Computing*, vol. 10, pp. 1129–1140, 2019.

[4] H. Zhang, G. Zhang, and Q. Yan, "Digital twin-driven cyber-physical production system towards smart shop-floor," *Journal of Ambient Intelligence and Humanized Computing*, vol. 10, pp. 4439–4453, 2019.

[5] A. Assad Neto, E. Ribeiro da Silva, F. Deschamps, and E. Pinheiro de Lima, "Digital twins in manufacturing: An assessment of key features," *Procedia CIRP*, vol. 97, pp. 178–183, 2021, 8th CIRP Conference of Assembly Technology and Systems. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2212827120314438

[6] Y. G. Kabaldin, D. A. Shatagin, M. S. Anosov, P. V. Kolchin, and A. M. Kuz'mishina, "Nc machine tools and digital twins," *Russian Engineering Research*, vol. 39, pp. 637–644, 2019.

[7] M. R. Nieto and R. B. Carmona-Benítez, "Arima + garch + bootstrap forecasting method applied to the airline industry," *Journal of Air Transport Management*, vol. 71, pp. 1–8, 2018.

[8] R. C. Chung, W. Ip, and S. Chan, "An arima-intervention analysis model for the financial crisis in china's manufacturing industry," *International Journal of Engineering Business Management*, vol. 1, p. 5, 2009.

[9] T. Sujjaviriyasup and K. Pitiruek, "A comparison between modwt-svm-de hybrid model and arima model in forecasting primary energy consumptions," in *2017 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*, 2017, pp. 799–802.

[10] G. Box, "Box and jenkins: time series analysis, forecasting and control," in *A Very British Affair*. Springer, 2013, pp. 161–215.

[11] "Time series forecasting using a hybrid arima and neural network model," *Neurocomputing*, vol. 50, pp. 159–175, 2003.

[12] T. Taskaya-Temizel and K. Ahmad, "Are arima neural network hybrids better than single models?" in *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005.*, vol. 5. IEEE, 2005, pp. 3192–3197.

[13] S. Panigrahi and H. Behera, "A hybrid ets–ann model for time series forecasting," *Engineering Applications of Artificial Intelligence*, vol. 66, pp. 49–59, 2017.

[14] P.-F. Pai and C.-S. Lin, "A hybrid arima and support vector machines model in stock price forecasting," *Omega*, vol. 33, no. 6, pp. 497–505, 2005.

[15] D. M. Izidio, P. S. de Mattos Neto, L. Barbosa, J. F. de Oliveira, M. H. d. N. Marinho, and G. F. Rissi, "Evolutionary hybrid system for energy

TABLE VII
PERCENTAGE IMPROVEMENT OVER THE ARIMA FOR THE DAILY AVERAGE SCENARIO.

Model	**Oil_Pressure**		**Coolant_Temp**		**Water_Temp**		**Oil_Temp**		**Redraw_Pressure**	
	MAPE	MSE	MAPE	MSE	MAPE	MSE	MAPE	MSE	MAPE	MSE
LSTM	**28.55**	40.34	3.17	35.03	6.15	2.62	-15.09	-20.93	-8.38	-9.47
SVR	2.65	2.27	**16.39**	41.3	**6.58**	-0.76	**5.05**	-1.40	-51.86	-95.74
MLP	9.27	10.3	5.58	**43.93**	6.33	0.18	-114.26	-193.45	-32.19	-75
XGBOOST	0.86	2.46	10.75	29.75	4.19	-1.92	-4.21	-5.11	-49.38	-90.24
ARIMA+MLP	3.19	12.13	-42.65	-42.83	-9.53	-8.06	-28.96	-20.16	-1.41	-14.05
ARIMA+XGB	22.08	39.28	10.75	3.07	-11.3	-7.59	1.8	0.3	-3.41	-4.83
ARIMA+SVR	-27.88	-67.58	16.39	21.1	-8.07	-0.74	0.95	0.18	-0.54	-1.13
ARIMA+LSTM	21.55	42.54	0.72	-3.51	-6.36	-4.44	-0.23	0.05	0.27	**1.03**
Proposed$_{\mathrm{Mean}}$	10.41	24.73	1.41	4.51	2.88	**5.75**	-3.2	0.65	-0.5	-1.39
Proposed$_{\mathrm{Mean}_2}$	22.75	**43.60**	3.04	18.36	1.84	3.15	1.53	0.31	-0.06	0.08
Proposed$_{\mathrm{Median}}$	15.28	32.36	2.03	0.82	-0.06	0.95	0.97	**0.68**	-0.85	-1.23
Proposed$_{\mathrm{OLS}_{(\mathrm{MLP},\mathrm{SVR})}}$	7.35	10.28	0.19	22.05	-17.18	-14.63	0.08	-1.45	-0.79	-1.04
Proposed$_{\mathrm{OLS}_{(\mathrm{MLP},\mathrm{XGBoost})}}$	-21.02	-48.36	- 0.7	21.53	1.39	2.06	-5.79	-3.19	-3.03	-5.46

consumption forecasting for smart meters," *Energies*, vol. 14, no. 7, p. 1794, 2021.

[16] P. S. de Mattos Neto, G. D. Cavalcanti, and F. Madeiro, "Nonlinear combination method of forecasters applied to pm time series," *Pattern Recognition Letters*, vol. 95, pp. 65–72, 2017.

[17] J. F. L. de Oliveira, E. G. Silva, and P. S. G. de Mattos Neto, "A hybrid system based on dynamic selection for time series forecasting," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–13, 2021.

[18] R. B. B. de Holanda and J. F. L. de Oliveira, "Swarm based ensembles for time series residual forecasting," in *2020 IEEE 32nd International Conference on Tools with Artificial Intelligence (ICTAI)*, 2020, pp. 595–602.

[19] M. Keijzer and V. Babovic, "Genetic programming, ensemble methods and the bias/variance tradeoff–introductory investigations," in *European Conference on Genetic Programming*. Springer, 2000, pp. 76–90.

[20] R. B. B. de Holanda and J. F. de Oliveira, "Swarm based ensembles for time series residual forecasting," in *2020 IEEE 32nd International Conference on Tools with Artificial Intelligence (ICTAI)*. IEEE, 2020, pp. 595–602.

[21] A. Kanawaday and A. Sane, "Machine learning for predictive maintenance of industrial machines using iot sensor data," in *2017 8th IEEE International Conference on Software Engineering and Service Science (ICSESS)*. IEEE, 2017, pp. 87–90.

[22] J.-G. Wang, Z. Xie, Y. Yao, B.-H. Yang, S.-W. Ma, and L.-L. Liu, "Soft sensor development for improving economic efficiency of the coke dry quenching process," *Journal of Process Control*, vol. 77, pp. 20–28, 2019.

[23] Y. Liu and J. Chen, "Integrated soft sensor using just-in-time support vector regression and probabilistic analysis for quality prediction of multi-grade processes," *Journal of Process Control*, vol. 23, no. 6, pp. 793–804, 2013.

[24] H. Kaneko and K. Funatsu, "Application of online support vector regression for soft sensors," *AIChE Journal*, vol. 60, no. 2, pp. 600–612, 2014.

[25] W. Ke, D. Huang, F. Yang, and Y. Jiang, "Soft sensor development and applications based on lstm in deep neural networks," in *2017 IEEE Symposium Series on Computational Intelligence (SSCI)*. IEEE, 2017, pp. 1–6.

[26] S. Soares, R. Araújo, P. Sousa, and F. Souza, "Design and application of soft sensor using ensemble methods," in *ETFA2011*. IEEE, 2011, pp. 1–8.

[27] X. Chen, Z. Mao, R. Jia, and S. Zhang, "Ensemble regularized local finite impulse response models and soft sensor application in nonlinear dynamic industrial processes," *Applied Soft Computing*, vol. 85, p. 105806, 2019.

[28] Y. Liu, D. Huang, B. Liu, Q. Feng, and B. Cai, "Adaptive ranking based ensemble learning of gaussian process regression models for quality-related variable prediction in process industries," *Applied Soft Computing*, vol. 101, p. 107060, 2021.

[29] X. Zhu, K. Hao, R. Xie, and B. Huang, "Soft sensor based on extreme gradient boosting and bidirectional converted gates long short-term memory self-attention network," *Neurocomputing*, vol. 434, pp. 126–136, 2021.

[30] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, 2016, pp. 785–794.

[31] M. Gurnani, Y. Korke, P. Shah, S. Udmale, V. Sambhe, and S. Bhirud, "Forecasting of sales by using fusion of machine learning techniques," in *2017 International Conference on Data Management, Analytics and Innovation (ICDMAI)*, 2017, pp. 93–101.

[32] D. Fan, H. Sun, J. Yao, K. Zhang, X. Yan, and Z. Sun, "Well production forecasting based on arima-lstm model considering manual operations," *Energy*, vol. 220, p. 119708, 2021.

[33] M. Khashei and M. Bijari, "A novel hybridization of artificial neural networks and arima models for time series forecasting," *Applied Soft Computing*, vol. 11, no. 2, pp. 2664–2675, 2011, the Impact of Soft Computing for the Progress of Artificial Intelligence.

[34] D. S. de O. Santos Júnior, J. F. de Oliveira, and P. S. de Mattos Neto, "An intelligent hybridization of arima with machine learning models for time series forecasting," *Knowledge-Based Systems*, vol. 175, pp. 72–86, 2019.

[35] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95-international conference on neural networks*, vol. 4. IEEE, 1995, pp. 1942–1948.

[36] A. J. Smola and B. Schölkopf, "A tutorial on support vector regression," *Statistics and computing*, vol. 14, no. 3, pp. 199–222, 2004.

[37] Y. Hua, Z. Zhao, R. Li, X. Chen, Z. Liu, and H. Zhang, "Deep learning with long short-term memory for time series prediction," *IEEE Communications Magazine*, vol. 57, no. 6, pp. 114–119, 2019.

[38] P. J. Adeodato, A. L. Arnaud, G. C. Vasconcelos, R. C. Cunha, and D. S. Monteiro, "Mlp ensembles improve long term prediction accuracy over single networks," *International Journal of Forecasting*, vol. 27, no. 3, pp. 661–671, 2011.

[39] N. Kourentzes, D. K. Barrow, and S. F. Crone, "Neural network ensemble operators for time series forecasting," *Expert Systems with Applications*, vol. 41, no. 9, pp. 4235–4244, 2014.

[40] R. J. Hyndman, Y. Khandakar *et al.*, "Automatic time series forecasting: the forecast package for r," *Journal of statistical software*, vol. 27, no. 3, pp. 1–22, 2008.

[41] R. J. Hyndman and A. B. Koehler, "Another look at measures of forecast accuracy," *International journal of forecasting*, vol. 22, no. 4, pp. 679–688, 2006.

[42] P. Goodwin and R. Lawton, "On the asymmetry of the symmetric mape," *International journal of forecasting*, vol. 15, no. 4, pp. 405–408, 1999.