27th ABCM International Congress of Mechanical Engineering
December 04-08, 2023, Florianópolis-SC, Brazil

COB-2023-1198
Fault detection in rotating machines using LSTM and time series analysis

Daniel Awada Elarrat Canto
Maurizio Radloff Barghouthi
Leonardo Kaucz
Eduardo Márcio de Oliveira Lopes
Giuliana Sardi Venter
Federal University of Paraná
daniel.awada@ufpr.br
giuliana.venter@ufpr.br

Abstract. *The need to reduce the downtime in machines has led to increased interest in process optimization in various fields, including mechanical engineering. One area of focus is predictive machine health analysis, which utilizes machine learning to predict when a serious failure may occur in a machine and minimize downtime. While predicting machine failures is a complex task, recent advances in machine learning have made this method more accessible and attracted growing interest from industries and researchers seeking to improve maintenance practices. This study focuses on the detection of faults in rotating machines caused by unbalance using machine learning. Experimental data was collected on a rotating machine test bench DAC VAD 203 under different conditions, with rotating speeds varying from 300rpm to 3300rpm and a forced unbalance faults ranging from 6.80 g to 20.4 g that, combined, resulted in 266 different experiments. Additional 242 experiments were synthetically generated by interpolating data with the same rpm and different forced unbalance, creating a new set of data that simulates the behavior of a machine when its unbalance intensity is changing. The data was collected using a triaxial accelerometer positioned on top of a selected ball bearing of a rotating system and then processed into time series by windowing the raw sensor data into equal time intervals of 0.5 seconds and extracting the time and frequency-domain features for each time segment, allowing the visualization of fault occurrences and analysis over time. A composed model consisted of a layer of a recurrent neural network and a regressor model was chosen and trained with the dataset in order o automate the task of fault and intensity detection. The recurrent model used was the Long-Short Term Memory (LSTM), as it has a good performance to make prediction based on time series. The model was tuned using an optimizer based on the Gradient Descent algorithm in Python, which iteratively adjusts the weights and biases of the model to minimize the loss function and improve the accuracy of its predictions. An margin of error lower than 6.8 g in unbalance is expected, with a short processing time, based on preliminary results. Comparing with other strategies, it is expected that this new method can significantly outperform previous approaches, highlighting the importance of tailored optimization strategies for complex machine learning models.*

Keywords: *Machine Learning, Rotating Dynamics, Synthetic Data, Process Monitoring, Recurrent Neural Networks*

1. INTRODUCTION

Rotating machines play a fundamental role in various industrial sectors, driving production processes and ensuring the proper functioning of equipment and operational systems. However, like any mechanical equipment, they are also susceptible to failures, which can compromise their efficiency and cause significant damage. Among the most common failures is the unbalance, characterized by the non-uniform distribution of mass along the rotor. The early detection of these failures is extremely important to prevent unplanned downtime, production losses and safety risks (**?**).

In this context, predictive maintenance has proven to be an effective approach in the management and maintenance of rotating machines. Unlike preventive maintenance, which is based on fixed time intervals, and corrective maintenance, which occurs only after failure, predictive maintenance uses data and advanced techniques for continuous monitoring of the machine's health, allowing for prediction and proactive action regarding potential failures (Schwendemann *et al.* (2021)). However, its implementation is still challenging due to its complexity and the high initial investments required for monitoring technologies and data analysis (Kawai *et al.* (2020)). In this context, the use of machine learning to monitor condition of production systems in general (Żabiński *et al.* (2019)) and of rotors especifically (Zhu *et al.* (2023)) emerges as a promising approach to tackle the complexity of the problem and provide efficient solutions.

Although the use of machine learning to predict faults in rotating machinery has been extensively researched, with applications ranging from using simple models such as Random Forests (Hu *et al.* (2020)) to very deep neural networks (Xu *et al.* (2022)), the analysis of the system in time series has been scarce. One complication of the analysis in time series resides in the dataset used to train the models, as it is necessary that the dataset change over time, which is both time consuming and difficult to obtain.

This article presents a detailed study on the application of machine learning for the detection of failures in rotating machines, with a focus on unbalance, in the time series. The proposed model is based on the combination of a Long Short-Term Memory (LSTM) networks, widely used in time series analysis, with a regression model, that aims to predict the machine unbalance based on the machine behavior. The use of time series data allows for capturing dynamic information about the machine behavior over time and making predictions about its future behavior, providing the regression model accurate estimates of unbalance in the future time. In order to do so, a time series synthetic data generation method is used.

Throughout the article, the details of the proposed model will be presented, as well as the methods used for data collection and preprocessing. After that, the results obtained through laboratory experiments will be discussed, comparing the performance of different regression models, such as Decision Tree, Random Forest, SVR, KNN, and the LSTM neural network. The analysis of the results will evaluate the effectiveness of the proposed model in identifying and analyzing failures in rotating machines.

The results obtained so far are promising, demonstrating the effectiveness of this method in accurately and reliably identifying faults in rotating machines. With this advancement, it is expected that the implementation of predictive maintenance in the industry will be more widespread and efficient, contributing to process optimization and ensuring operational safety.

2. METHODS

2.1 Data gathering

To obtain the data necessary for the training of machine learning models, a series of experiments was conducted in a controlled laboratory. The experiment involved using a rotating machine, a dynamic signal acquisition module, a triaxial accelerometer, weights to induce unbalance in the machine and software for data visualization.

The variables parameters in the experiments were the rotations per minute of the machine and the level of the induced unbalance. The rotations per minute was adjusted in the range of 300 to 3300, with increments of 300. As for the unbalance, four levels were used: 0.00 g (normal state), 6.80 g, 13.60 g, and 20.4 g of forced unbalance was induced by adding weights to each of the two discs of the system. The experiments were conducted with different combinations of weights, ranging from 0 to 3 weights on each disc.

Each experiment had a duration of 5 seconds, with the exception of the experiments with 0 g of unbalance, that have 1 minute each to balance the amount of data in normal state and in fault state. For each experiment the acceleration data were collected on the three axes (X, Y, and Z) of the triaxial accelerometer, along with the corresponding timestamp. The sampling frequency used was 1000 Hz, ensuring a high temporal resolution in data capture.

The use of this laboratory method enabled the acquisition of a diverse and controlled dataset, ranging different combinations of RPM and unbalance levels. This data is essential for the training and validation of the machine learning models, enabling precise and efficient analysis in detecting faults in rotating machines.

2.2 Time series synthetic data generation

In order to predict the fault in time series, it is necessary to create a dataset that changes considerably in time. Although it would be possible to create such a dataset by collecting data for a long period of time, it would be cost prohibitive. Therefore, the data on the machine's behavior during the transition from its normal state to an arbitrary unbalanced state - or with variations on the intensity of this unbalance - is generated trough a time series synthetic data generation. To achieve this, additional data was added to the dataset by interpolating data with different unbalance intensities and simulating the machine's behavior in these scenarios.

The data interpolation was performed linearly over time between two or more time series. For each quantity of interpolated series, the intensity of each one and the visualization of the used windowing can be seen in Figure 1. The sum of all windows is always constant and equal to 1 due to the proportionality in the decrease of one window in relation to the increase of the others, ensuring conservation of the signal's amplitude and energy during the interpolation (Sestito *et al.* (2022)).

The interpolations were only performed between time series with the same rotations per minute and with unbalance intensity ordered in ascending order. Thus, considering all possible combinations of different unbalances and rotations, 242 new synthetic experiments were generated in the database, improving the quality of the data and the training of the machine learning models.

2.3 Pre-processing

To emphasize the essential properties of the data and improve the quality of the model training, a pre-processing module was performed in two steps. The first step involved removing frequencies with low amplitude, where any frequency

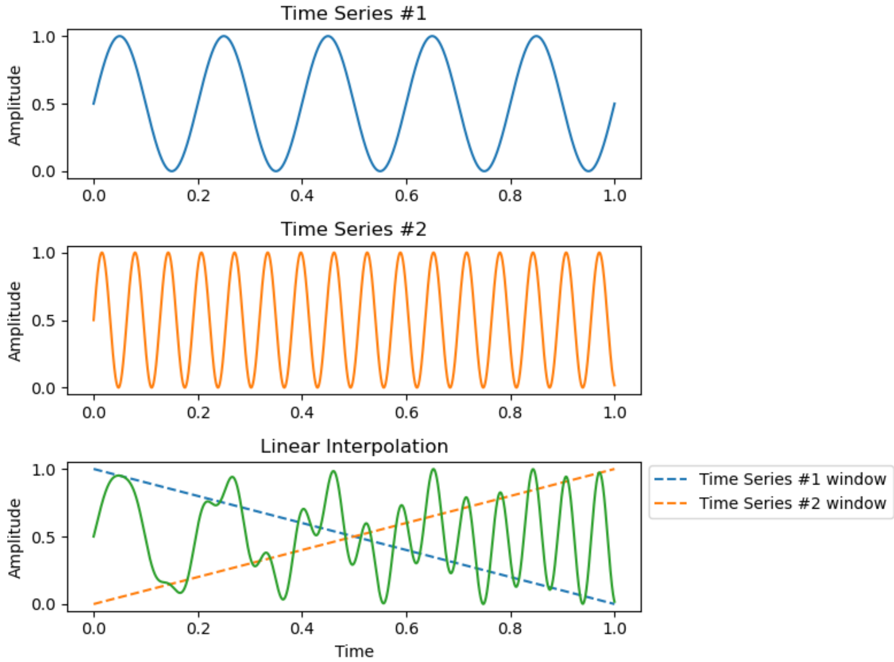

Figure 1. Example of a linear interpolation between 2 time series

with an amplitude less than 5% of the peak was considered noise and its amplitude was reduced to 0. In the second step, isolated frequency with an amplitude equal or less than 20% of the peak is identified and have its amplitude also reduced to 0. Where 'isolated frequency' means a frequency with the value of its adjacent frequencies equals to 0.

Figure 2 illustrates an example signal at each stage of the pre-processing, from the transformation to the frequency domain using the Fast Fourier Transform to the final signal reconstruction. The ultimate goal of this pre-processing is to remove a significant portion of the noise present in the signals and emphasize relevant vibration frequencies.

2.4 Feature Extraction

After completing the processing, it is necessary to extract the features that will be used in the prediction model. For this purpose, the time series are synthesized into unique values that represent different properties of the machine's behavior, both in the time domain and in the frequency domain. The chosen properties were defined independently for each of the X, Y, and Z axes, resulting in a total set of 48 features.

The definitions and calculations of the properties can be found in Table 1, where the symbol N represents the number of data points in the series, x_i represents a given value at a time instant, and X_i is a the absolute value of an amplitude in the frequency domain. These features are calculated using aggregation functions applied to the original time series, allowing the capture of relevant information for fault detection.

These features were chosen for its ability of summarizing the main characteristics of a signal as a whole. They represent the system's behavior in various different aspects, where each one of them may influence the target in a different way. By using these features as inputs, it is found suitable ways to adjust a machine in order to achieve the desired target Ribeiro (2018).

2.5 Outliers Removal

To ensure the quality of the data used for training the machine learning model, this study includes a custom outlier removal method. The outliers are identified calculating the distances between the maximum and minimum points with respect to the series mean and them compared with the 99th and 1st percentiles. If the ratio between the distances is equal to or greater than 2, the point is considered an outlier and is excluded from the database. The impact of outlier removal is visualized in the Figure 3, showing the series mean as the black line, the 99th and 1st percentiles as the orange lines, and the maximum and minimum as the green lines. Initially the dataset with all the features extracted both by the experimental data and the synthetic data was consisted by 20,900 samples with 823 of them being considered outliers and removed from the dataset, a reduction of 3.94%. This outlier removal process is crucial in preparing the data for regression model training, improving the quality of the training set and enhancing the model's performance (Theissler *et al.* (2021)).

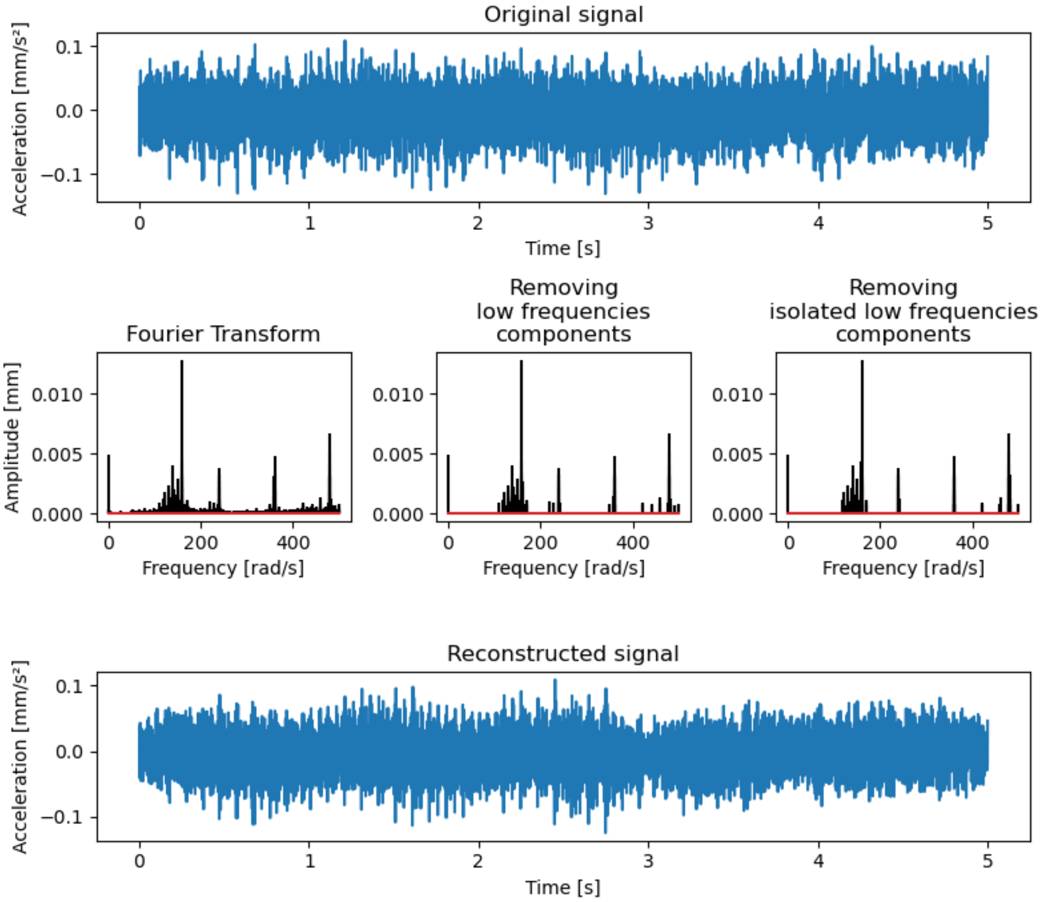

Figure 2. Visualization of every step of the pre-processing

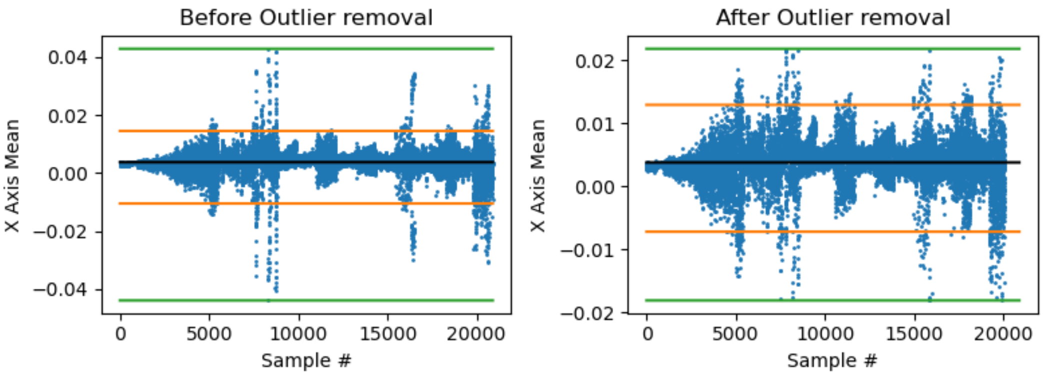

Figure 3. Before and after the Outlier Removal step for one of the features

2.6 Feature Selection

To reduce the number of features and speed up the method for both training and implementation of the final model, two filters will be applied: the information quality filter and the correlation filter. The first filter, the informational quality of a feature, measures the ratio between the standard deviation within a class and the standard deviation between classes. If the data within a class for a feature are all similar but different from the other classes, that feature will be considered highly informative; otherwise, it will be considered less informative.

Furthermore, having features with high correlation hinders training because it gives more weight to certain properties than others. Therefore, the next step in the feature selection is to remove those with high correlation. The chosen threshold value is 0.8, meaning that if two features have a correlation equal or greater than this value, one of them will be disregarded. This filter will return 20 final features that will be considered in the final training of the models, which are presented in Table 2.

Table 1. Features definitions

Feature	**Symbol**	**Definition**
Mean	μ_x	$\mu_x = \frac{1}{N}\sum_{i=0}^{N} x_i$
Standard Deviation	σ_x	$\sigma_x^2 = \frac{1}{N}\sum_{i=0}^{N}(x_i - \mu_x)^2$
Kurtosis	κ_x	$\kappa_x = \frac{1}{N}\sum_{i=0}^{N}\left(\frac{x_i-\mu_x}{\sigma_x}\right)^4$
Skewness	γ_x	$\gamma_x = \frac{1}{N}\sum_{i=0}^{N}\left(\frac{x_i-\mu_x}{\sigma_x}\right)^3$
Peak-to-Peak Amplitude	x_{ppv}	$x_{ppv} = max(x_i) - min(x_i)$
Root Mean Square	x_{rms}	$x_{rms} = \left(\frac{1}{N}\sum_{i=0}^{N} x_i^2\right)^{1/2}$
Square Root Amplitude	x_{sra}	$x_{sra} = \left(\frac{1}{N}\sum_{i=0}^{N}\sqrt{\lvert x_i\rvert}\right)^2$
Crest Factor	x_{cf}	$x_{cf} = \frac{max(\lvert x_i\rvert)}{x_{rms}}$
Impulse Factor	x_{if}	$x_{if} = \frac{max(\lvert x_i\rvert)}{\frac{1}{N}\sum_{i=0}^{N}\lvert x_i\rvert}$
Margin Factor	x_{mf}	$x_{mf} = \frac{max(\lvert x_i\rvert)}{x_{sra}}$
Kurtosis Factor	x_{kf}	$x_{kf} = \frac{\kappa_x}{x_{rms}^4}$
Mean (FFT)	μ_x	$\mu_X = \frac{1}{N}\sum_{i=0}^{N} X_i$
Standard Deviation (FFT)	σ_x	$\sigma_X^2 = \frac{1}{N}\sum_{i=0}^{N}(X_i - \mu_X)^2$
Root Mean Square (FFT)	x_{rms}	$x_{rms} = \left(\frac{1}{N}\sum_{i=0}^{N} X_i^2\right)^{1/2}$
Peak Value (FFT)	X_{max}	$X_{max} = max(X)$
Peak frequency (FFT)	f_{pico}	$f_{pico} = f(X_{max})$

Table 2. Selected features

No.	**Axis**	**Feature**
1	X Axis	Mean
2	X Axis	Peak-to-Peak Amplitude
3	X Axis	Kurtosis
4	X Axis	Skewness
5	X Axis	Kurtosis Factor
6	X Axis	Mean (FFT)
7	Y Axis	Mean
8	Y Axis	Kurtosis
9	Y Axis	Skewness
10	Y Axis	Kurtosis Factor
11	Y Axis	Impulse Factor
12	Y Axis	Peak Value (FFT)
13	Y Axis	Mean (FFT)
14	Z Axis	Mean
15	Z Axis	Kurtosis
16	Z Axis	Skewness
17	Z Axis	Kurtosis Factor
18	Z Axis	Margin Factor
19	Z Axis	Mean (FFT)
20	Z Axis	Peak Value (FFT)

An additional feature also considered is the rpm, due to its importance for unbalance prediction, totaling 21 resources in total.

2.7 LSTM training

The LSTM model has the objective to predict the features values in the future based on its historic values (Ma *et al.* (2021)). Before being used in the LSTM model, the signal is normalized relative to the test data, ensuring that all features are in the range of values from 0 to 1. A single LSTM model is built to predict all features at once, with one recurrent layer containing 16 units and an output dense layer with 21 units corresponding to the 21 input features. The model uses the Mean Squared Error as its loss function, is trained with 36 samples for 100 epochs and a batch size of 16. It is important to remember that these data points form a time series, as shown in the Figure 4. Given the features for each chunk of a signal, the LSTM model should be capable to predict the feature of the next chuck in the future

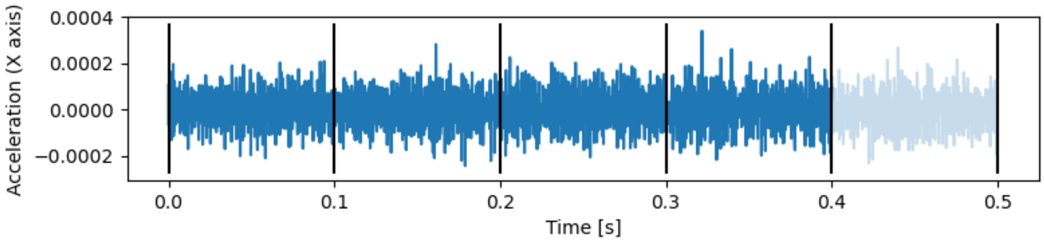

Figure 4. Time series associated to the LSTM model

This LSTM model is capable of making predictions for a time horizon of 0.1 seconds. This prediction can be performed iteratively to project further into the future, but compromising the accuracy of the predictions. The metric used to evaluate the performance of the LSTM model is Mean Squared Error (MSE), which measures the average squared error between the model's predictions and the actual values.

2.8 Regressor model training

To find the regression model that accurately estimates the unbalance of a rotating machine, five different model options were explored: Decision Tree, Random Forest, Support Vector Regression (SVR), K-Nearest Neighbors (KNN), and Artificial Neural Network (ANN). An hyperparameter tuning was performed for each model with the intention of finding the best possible model for the problem.

The MSE will serve as the key metric for selecting the best regressor model to be combined with the LSTM layer, resulting in the final model. A lower MSE value indicates superior regression performance, making it the primary criterion for model selection. Detailed explanations of the MSE and other relevant metrics can be found in the subsequent section.

2.9 Model validation

To evaluate the errors in regression tasks, three metrics were used: the coefficient of determination R-Squared (R^2) which represents how well the model represents the dataset, given by the Eq. 1, the Mean Absolute Error (MAE), which represents the margin of error of the predictor, given by the Eq. 2, and the Mean Squared Error (MSE), that represents how well the predicted data fits the real data, given by the Eq. 3.

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(Z_{s,i} - \hat{Z}_{s,i})^2}{\sum_{i=1}^{n}(Z_{s,i} - \bar{Z}_s)^2}, \tag{1}$$

$$MAE = \frac{\sum_{i=1}^{n} \mid \hat{Z}_{s,i} - Z_{s,i} \mid}{n}, \tag{2}$$

$$MSE = \frac{\sum_{i=1}^{n}(\hat{Z}_{s,i} - Z_{s,i})^2}{n}, \tag{3}$$

in which n is the number of datapoints on the dataset, $Z_{s,i}$ is the actual target value, $\hat{Z}_{s,i}$ is the predicted target value, and $\bar{Z}_s$ is the average of the target values.

All the models were trained and tested with a 0.2 train test split, in which 20% of the dataset is randomly separated to be used as a validation dataset.

3. RESULTS

3.1 LSTM model

Performance evaluation of the LSTM model was made based on three metrics: Mean Squared Error (MSE), Mean Absolute Error (MAE), and Coefficient of Determination (R2). The values for each metric are presented in the Table 3.

Table 3. LSTM Model errors metrics

Metric	Value
MSE	3.1868e-3
MAE	3.3125e-2
R2	0.634221

These results highlight the effectiveness of the LSTM model for this proposed method, providing accurate and reliable behavior predictions based on the input features. The low error rates, both in terms of MSE and MAE, indicate a good generalization ability of the model for unseen data. Additionally, the significant R2 value demonstrates the model's capacity to explain a considerable portion of the variability in the test data.

The obtained results reinforce the relevance and applicability of the LSTM model in the project's context, providing valuable insights for understanding and predicting the desired behavior. These findings contribute to the advancement of the machine learning field and emphasize the feasibility and effectiveness of the approach adopted in this study.

3.2 Regressor model

The training results for each of the five models are presented in Table 4 and the model with the lowest Mean Squared Error (MSE) score was considered the best one. The table shows that the Random Forest model obtained the lowest MSE value (9.13) compared to the others, indicating it is the most efficient model for estimating the unbalance of a rotating machine.

The Random Forest also had the lowest Mean Absolute Error (MAE) values and the highest coefficient of determination (R2), demonstrating its ability to make more accurate predictions on the unbalance of a rotating machine and explain a greater percentage of the data variability.

Table 4. Regression Models comparison

Metric	MSE	MAE	R2
Decision Tree	13.286442	2.383776	0.722186
Random Forest	9.126615	2.118423	0.809166
SVR	19.371223	3.156162	0.594955
KNN	11.956285	2.223472	0.749999
Nural Network	17.728253	2.984338	0.629309

It is important to note that the unbalance ranged from 0 to 20.4 grams during the experiments, with increments of 6.8 grams. Therefore, the Random Forest was able to predict the unbalance with an average precision of 2.12 grams, which represents an absolute variation of only 31.2% compared to the smallest unit of unbalance. This indicates that the chosen model is effective in estimating the unbalance in rotating machines, aiding in the identification and prevention of failures and contributing to predictive maintenance and the improvement of operational efficiency.

3.3 Final model

After training the individual LSTM Predictor and regression models, a pipeline was created to perform the analysis and prediction of unbalance in rotating machines. The pipeline is presented in Figure 5.

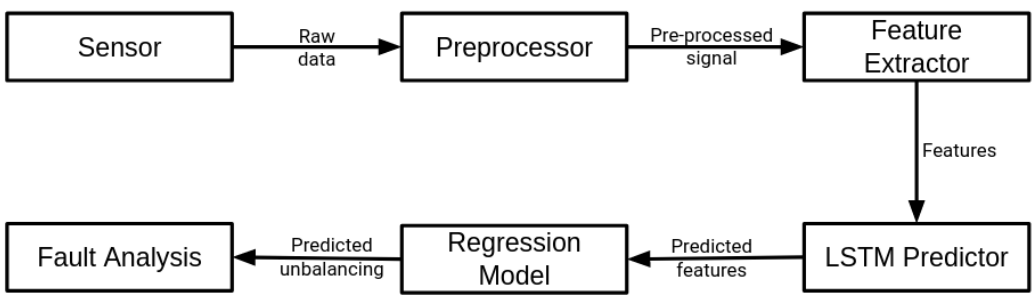

Figure 5. Methodology flowchart

In this model, the LSTM function is to predict future values of features based on historical data and then use these predicted features as input to the regression model, which estimates the unbalance. Finally, the predicted result is compared with acceptable unbalance values in the fault analysis step, generating reports and issuing alerts in case of significant failures. The results obtained with the complete model can be seen in Table 5.

Table 5. Final Model errors metrics

Metric	Value
MSE	25.8794
MAE	3.9852,
R2	0.3781

This final machine learning model, made by a combination of a LSTM Predictor and a regressor, has shown the ability to make accurate predictions and estimates of unbalance in rotating machines. Considering that the unbalance ranged from 0 to 20.4 grams during the experiments, an MAE of 3.99 indicates that the model has an absolute variation smaller than the smallest increment of unbalance, which is considered an satisfactory margin of error for efficient and secure implementation.

4. CONCLUSIONS

In conclusion, this study successfully applied machine learning techniques, combining a LSTM model with an regression model, for the detection of unbalance in rotary machines. By utilizing time series data, applying the necessary preprocessor and extracting relevant features, the proposed model demonstrated accurate predictions and estimations of machine unbalance. The results showed low errors in terms of MSE and MAE, indicating the model's ability to generalize well to unseen data. Additionally, the high coefficient of determination (R2) highlighted the model's capability to explain a considerable portion of the variability in the test data. The combination of LSTM and regression proved to be effective in forecasting and estimating machine unbalance, enabling proactive identification and analysis of faults. This approach contributes to the advancement of machine learning in the field of predictive maintenance and enhances operational efficiency by preventing unplanned downtime and minimizing potential damages. The findings from this study emphasize the relevance and applicability of the proposed model, providing valuable insights for understanding and predicting machine behavior accurately.

5. ACKNOWLEDGEMENTS

Eduardo M. O. Lopes acknowledges the support of CNPq - National Council for Scientific and Technological Development for his scholarship, as Maurizio R. Barghouthi and Daniel A. E. Canto acknowledge the support of CAPES - Coordination for the Improvement of Higher Education Personnel for their scholarship.

6. REFERENCES

Hu, Q., Si, X.S., Zhang, Q.H. and Qin, A.S., 2020. "A rotating machinery fault diagnosis method based on multi-scale dimensionless indicators and random forests". *Mechanical Systems and Signal Processing*, Vol. 139, p. 106609. ISSN 0888-3270. doi:https://doi.org/10.1016/j.ymssp.2019.106609. URL `https://www.sciencedirect.com/science/article/pii/S0888327019308301`.

Kawai, K., Irino, N., Iiyama, K., Matsubara, A., Mizuyama, H., Mori, K. and Fujishima, M., 2020. "A prediction model for high efficiency machining conditions based on machine learning". Elsevier B.V., Vol. 101, pp. 54–57. ISBN 9781713835431. ISSN 22128271. doi:10.1016/j.procir.2020.09.188.

Ma, X., Xie, X., Li, X., Li, X. and Li, J., 2021. "Research on the classification algorithm of elevator faults based on machine learning". *Applied Sciences*, Vol. 11, No. 7, p. 3073. doi:10.3390/app11073073.

Ribeiro, F.M.L., 2018. "Similarity-based methods for machine diagnosis".

Schwendemann, S., Amjad, Z. and Sikora, A., 2021. "A survey of machine-learning techniques for condition monitoring and predictive maintenance of bearings in grinding machines". doi:10.1016/j.compind.2020.103380.

Sestito, G.S., Venter, G.S., Ribeiro, K.S.B., Rodrigues, A.R. and da Silva, M.M., 2022. "In-process chatter detection in micro-milling using acoustic emission via machine learning classifiers". *International Journal of Advanced Manufacturing Technology*, Vol. 120, pp. 7293–7303. ISSN 14333015. doi:10.1007/s00170-022-09209-w.

Theissler, A., Pérez-Velázquez, J., Kettelgerdes, M. and Elger, G., 2021. "Predictive maintenance enabled by machine learning: Use cases and challenges in the automotive industry". *Reliability Engineering and System Safety*, Vol. 215. ISSN 09518320. doi:10.1016/j.ress.2021.107864.

Xu, H., Liu, J., Peng, X., Wang, J. and He, C., 2022. "Deep dynamic adaptation network: a deep transfer learning framework for rolling bearing fault diagnosis under variable working conditions". *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, Vol. 45, No. 1, p. 41. ISSN 1806-3691. doi:10.1007/s40430-022-03950-9. URL `https://doi.org/10.1007/s40430-022-03950-9`.

Zhu, Z., Lei, Y., Qi, G., Chai, Y., Mazur, N., An, Y. and Huang, X., 2023. "A review of the application of deep learning in intelligent fault diagnosis of rotating machinery". *Measurement*, Vol. 206, p. 112346. ISSN 0263-2241. doi:https://doi.org/10.1016/j.measurement.2022.112346. URL `https://www.sciencedirect.com/science/article/pii/S0263224122015421`.

Żabiński, T., Mączka, T., Kluska, J., Madera, M. and Sęp, J., 2019. "Condition monitoring in industry 4.0 production systems - the idea of computational intelligence methods application". *Procedia CIRP*, Vol. 79, pp. 63–67. ISSN 2212-8271. doi:https://doi.org/10.1016/j.procir.2019.02.012. URL `https://www.sciencedirect.com/science/article/pii/S221282711930126X`. 12th CIRP Conference on Intelligent Computation in Manufacturing Engineering, 18-20 July 2018, Gulf of Naples, Italy.

7. RESPONSABILITY NOTICE

The authors are the only responsible for the printed material included in this paper.