engineering proceedings

Proceeding Paper

Extracting Running-in Dynamics from Operational Time Series Using Long Short-Term Memory Networks †

Theodor R. van Caspel [1], Gabriel Thaler [2,*] and Rodolfo C. C. Flesch [2]

[1] Department of Mechanical Engineering, Universidade Federal de Santa Catarina, Florianópolis 88040-900, SC, Brazil; theodorcaspel@gmail.com
[2] Department of Automation and Systems Engineering, Universidade Federal de Santa Catarina, Florianópolis 88040-900, SC, Brazil; rodolfo.flesch@ufsc.br
* Correspondence: thaler.gabriel@posgrad.ufsc.br
† Presented at the 12th International Conference on Time Series and Forecasting, Canaria, Spain, 15–17 July 2026.

Abstract

The running-in phase of hermetic reciprocating compressors is characterized by slow, nonstationary changes in operational signals and is commonly assessed using empirically defined test durations or manually defined indicators. From a data-analysis perspective, this constitutes a feature extraction problem, as informative temporal patterns might be identifiable from long and noisy time series acquired during operation. This paper investigates the use of Long Short-Term Memory (LSTM) networks as a data-driven feature extraction mechanism for running-in analysis based on electrical current and vibration measurements. LSTM-based sequence encoding is used to transform raw time-series segments into compact representations that summarize their temporal structure and enable discrimination between running-in and steady-state regimes, as well as characterization of intermediate conditions in the learned feature space using a multilayer perceptron. Model structure, input configuration, and segmentation parameters are selected through an optimization procedure under weak supervision. Experimental results show that the extracted LSTM features capture consistent temporal signatures of running-in across different compressor units, with one unit exhibiting outlier behavior in latent space analysis. The results support the use of recurrent neural networks for noninvasive running-in assessment in rotating machinery and related condition monitoring tasks.

Keywords: LSTM; feature extraction; running-in; hermetic reciprocating compressors

1. Introduction

Time series measurements acquired from industrial systems often exhibit informative temporal structures within short observation windows. In such cases, relevant information is encoded in the ordering and dynamics of the signal rather than in isolated samples or static summary statistics. Models that can extract features from finite-length sequences without relying on handcrafted descriptors are therefore of interest in a wide range of applications [1–3]. This setting is typical in industrial testing scenarios, where high-frequency measurements are acquired periodically and analyzed as independent realizations of the system behavior. High-frequency data are often recorded over short time intervals in regular time steps [4]. Although successive measurements may be treated independently, the temporal structure within each window can still reflect underlying conditions.

Academic Editors: Olga Valenzuela, Fernando Rojas, Luis Javier Herrera, Hector Pomares and Ignacio Rojas

Published: 18 September 2026

https://doi.org/10.3390/engproc2026155010

One application in which such analysis is relevant is the assessment of the running-in phase of mechanical systems, which corresponds to the initial stage of their service life. Although the literature describes methods for modeling this phenomenon, most studies focus on simplified tribological systems rather than complex machinery [5,6]. A relevant case study of running-in estimation and modeling is in hermetic reciprocating compressors, as the phenomenon affects frictional conditions and internal losses, and its correct identification is required by international standards to ensure repeatable performance tests [7,8]. Due to the sealed construction of these devices, the completion of running-in is usually inferred using empirically defined operating times, which do not account for variability between compressor units [9].

Some classic works have proposed the indirect estimation of running-in using statistical features extracted from the current and vibration in machinery [10,11]. However, due to the many moving parts and the nature of compressor operation, the influence of running-in on these signals may be subtle and may not appear as a distinct event, but rather as changes in the temporal structure of the measured time series [12,13]. Some recent studies have proposed using data-driven methods, such as unsupervised ensembles and semi-supervised classifiers, but these methods are still based on handcrafted statistical features and, as such, are not able to capture the temporal dynamics of the measured data.

Long short-term memory (LSTM) networks are commonly used for sequence modeling and feature extraction in time series analysis [14]. By means of gated recurrent units, LSTMs can learn temporal dependencies within fixed-length sequences and produce compact representations of their dynamic behavior. When applied to independent time series segments, LSTMs act as nonlinear temporal feature extractors, capturing short- and medium-range dependencies without assuming long-term temporal continuity between successive segments [15]. These networks are used in state-of-the-art methods for evaluating health and behavior in complex machinery, suggesting a high potential for extracting features from measurements that might help in modeling the running-in behavior of reciprocating compressors [2,3,16].

In this work, an LSTM-based framework is proposed to extract running-in-related features from operational time series acquired during compressor tests. Each measurement window is treated as an independent multivariate time series and processed by an LSTM network to learn features that describe its internal temporal dynamics. A weakly supervised formulation with a multilayer perceptron (MLP) is adopted, in which only the early and late operating periods are labeled, allowing the model to associate changes in within-window dynamics with progression toward steady-state operation. The proposed approach provides a data-driven alternative to empirically defined running-in durations and is suitable for automated testing and condition assessment applications.

2. Materials and Methods

This section describes the experimental data, the proposed time-series processing and learning framework, and the methodology adopted for training, optimization, and evaluation. An overview of the pipeline is shown in Figure 1.

The method takes as input time-series measurements obtained from experimental tests. These data are partially labeled according to guidelines from the manufacturer regarding the expected minimum and maximum duration of the running-in process. The output, p_{r}, is the probability of running-in given the input measurements. The experimental test rig and procedure are described in Section 2.1, along with the prelabeling of the experimental dataset.

The estimation process comprises two steps: the preprocessing step, which adjusts the length and sampling rate of the time-series, and the estimation step, which includes the LSTM for time-series feature extraction and the MLP for steady-state estimation. These steps are further described in Sections 2.2 and 2.3, respectively.

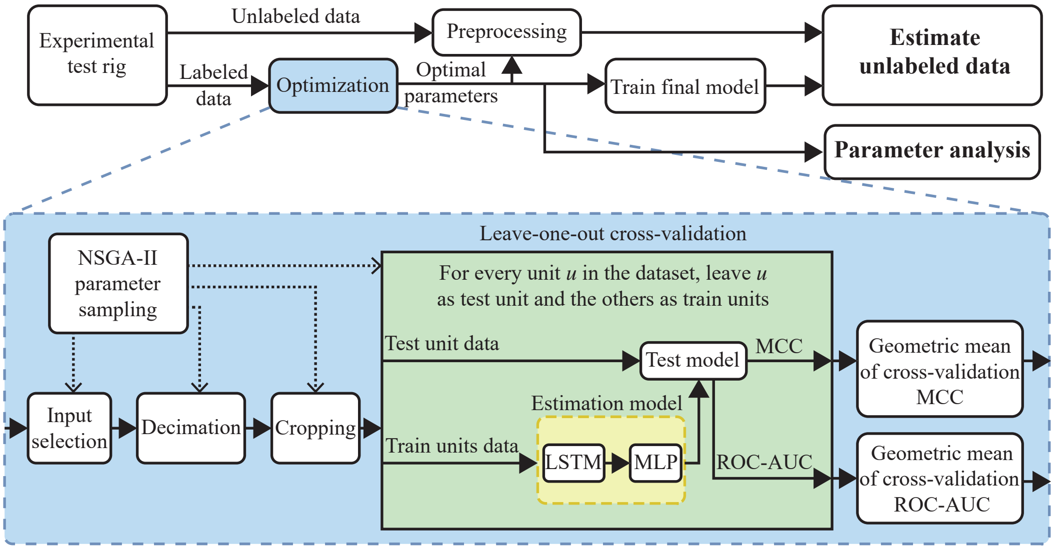

Figure 1. Proposed framework for feature extraction and running-in estimation.

The data processing parameters and hyperparameters for the LSTM and MLP were optimized for the running-in estimation problem using the Non-dominated Sorting Genetic Algorithm II (NSGA-II) [17]. The metrics chosen as the objective for optimization are described in Section 2.4, as well as further descriptions of the complete framework for optimizing, training, and evaluating the model.

2.1. Experimental Data and Problem Setup

The experimental data used in this work were obtained from a controlled test-rig for running-in tests performed on hermetic reciprocating compressors for refrigeration applications, shown in Figure 2a,b. During these tests, the compressors were operated under constant suction and discharge pressures and temperature conditions to ensure that the variations observed in the measured signals could be primarily attributed to the evolution of the tribological state rather than to external disturbances. The variables of interest were continuously monitored using noninvasive instrumentation, including the high-frequency measurement of the lateral vibration in the lower half of the compressor shell ($\boldsymbol{V}_{\mathrm{lat}}$), the longitudinal vibration in the upper half of the compressor shell ($\boldsymbol{V}_{\mathrm{long}}$), and the electrical current of the compressor motor ($\boldsymbol{I}$), all at a sampling rate of 25.6 kHz. The instrumented compressor is shown in Figure 2c. Every minute, a 1 s time-series was acquired for each of the high-frequency measurements, providing the inputs for the proposed estimation framework.

The running-in phenomenon manifests itself as a gradual and nonstationary transition that occurs only once during the early life of each compressor unit. Consequently, reliable labels for the running-in state are only partially available. In this work, a semi-supervised labeling strategy was adopted: data acquired during the initial period of operation were assumed to correspond to the running-in regime, while data acquired after an extended period of operation were assumed to represent the tribological steady-state. Measurements collected during the intermediate period were treated as unlabeled and used exclusively for inference and evaluation.

To assess the generalizability of the proposed method, data from 4 compressor units of the same compressor model were tested in the experimental setup proposed in this work. Three tests of 25 h were performed with each compressor unit, the first being the running-in test, and the subsequent ones being reference tests with the device already run-in. Based on information provided by the manufacturer, it was assumed as a ground truth that the running-in procedure could take between 5 h and 24 h, so the samples between these limits were left unlabeled. In order to capture general running-in dynamics rather than unit-specific characteristics, a leave-one-compressor-out cross-validation strategy was employed during optimization, which is further described in Section 2.4.

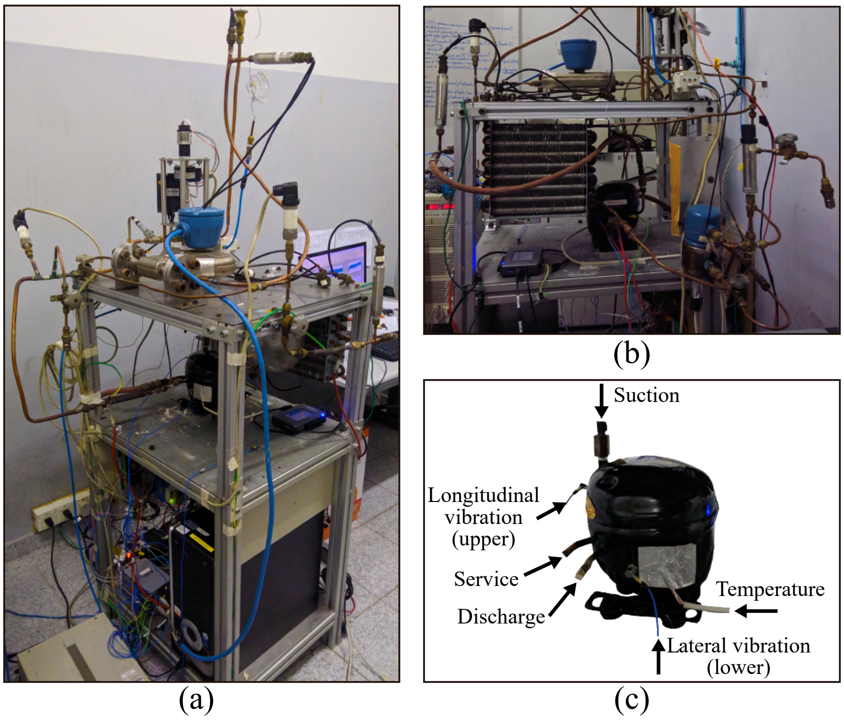

Figure 2. Pictures of the running-in test rig: (**a**) perspective view; (**b**) lateral view of the intermediary stage; (**c**) instrumented compressor.

2.2. Time-Series Preprocessing

The raw operational signals consist of high-frequency time series acquired over several hours of compressor operation. Directly processing these signals at their original sampling rate would lead to excessive computational costs and redundancy. Therefore, a preprocessing stage was applied before feature extraction.

The preprocessing stage consisted of input selection, signal decimation to reduce the effective sampling rate while preserving dominant temporal patterns, and segmentation into fixed-length subsequences. Each subsequence represents a temporal window of compressor operation and constitutes one input sample for the learning model.

The decimation factor and the length of the subsequence play a significant role in determining the temporal resolution and contextual information available to the model [18,19]. Rather than fixing these parameters heuristically, they were treated as hyperparameters and optimized jointly with the model architecture, as described in Section 2.4.

2.3. LSTM-Based Time-Series Feature Extraction and Running-in Estimation

To automatically extract representative temporal features from the preprocessed subsequences, a multilayer LSTM network was employed. Each input subsequence was processed by a multilayer LSTM, which maps the sequence to a latent feature vector at the final time step, interpreted as a compact representation of its temporal dynamics.

The latent feature vectors extracted by the LSTM network were subsequently used as inputs to an MLP responsible for estimating the running-in state. Instead of producing a binary classification, the MLP outputs a continuous value between zero and one, interpreted as the probability that the compressor has reached its tribological steady state.

This probabilistic formulation provides a smooth characterization of the running-in progression over time and enables the identification of transitional regimes that cannot be adequately described using hard class boundaries. The MLP consists of one or more fully connected hidden layers followed by a sigmoid output layer.

The number of LSTM and MLP layers, the dimensionality of the LSTM hidden state, the number of neurons per MLP layer, the activation function, and the use of regularization mechanisms, such as dropout, were treated as tunable hyperparameters in the optimization process. The framework for optimization, training, and evaluation of the models is further described in Section 2.4.

2.4. Training, Optimization, and Evaluation Framework

The training and evaluation of the model followed the framework illustrated in Figure 1. Only the labeled portions of the training dataset, corresponding to the early running-in period and the late steady-state period, were used to compute the loss function during training. Unlabeled data were excluded from the optimization process and were used solely for inference and qualitative analysis of the estimated running-in progression.

Hyperparameter optimization was performed using the NSGA-II, with the objective of jointly optimizing preprocessing parameters (which measurements to use as input, decimation factor, and subsequence length) and model hyperparameters (LSTM and MLP architecture). The Matthews correlation coefficient (MCC) was selected as the primary performance metric due to its balanced nature and robustness under class imbalance [20,21]. As a correlation-based measure, MCC has been shown to offer a more faithful summary of classifier performance than accuracy or precision-based metrics, particularly in imbalanced or semi-supervised classification problems [22,23].

The optimization process was initialized with a set of 50 random samples from the hyperparameter space, which were used in the preprocessing of the data and in the training of the classifier models. Being a genetic algorithm, at each iteration of the NSGA-II, new candidate configurations are generated through crossover and mutation operators applied to the best-performing configurations of the previous generation. These new candidates are subsequently evaluated and ranked based on Pareto dominance concerning the two objective metrics. The non-dominated sorting mechanism ensures that the algorithm maintains a diverse population of solutions, iteratively advancing the Pareto front to identify the optimal tuning.

To promote robustness across compressor units, model performance was evaluated using a leave-one-compressor-out cross-validation scheme, and the geometric mean of the validation folds was adopted as a conservative aggregation strategy. As MCC depends on a classification threshold, the threshold maximizing the geometric mean across folds was selected. Under this formulation, negative or undefined MCC values, which may arise for poorly performing classifiers or in degenerate confusion matrix configurations, would render the aggregated objective ill-defined. Therefore, MCC values that were negative or undefined for a given fold were set to zero prior to aggregation, leading to a zero value for the objective in case any classification was negative or undefined. In order to still provide a metric for advancing the optimization problem in such cases, the area under the receiver operating characteristic curve (ROC-AUC) was introduced as a complementary optimization objective. Unlike the MCC, ROC-AUC is always well-defined and bounded in the [0, 1] interval, providing a threshold-independent measure of class separability and ensuring the numerical stability of the optimization process while still preserving sensitivity to discriminative performance [22,24].

The optimization was run for 5000 iterations, after which the trials in the Pareto front were retrained using the full labeled portion of the training dataset and subsequently applied to the unlabeled data. The probability trajectories were analyzed to assess the model ability to capture the temporal evolution of the running-in process and the latent space generated by the LSTM network, and the results of the analysis are presented in Section 3.

3. Results and Discussion

Figure 3 presents the 5000 optimization trials and the geometric mean of the cross-validation metrics obtained with the labeled portion of the experimental data described in Section 2.1. The trials in the Pareto front, highlighted in the figure, are also listed in Table 1 along with the corresponding performance metrics. The highest geometric means of cross-validation ROC-AUC and MCC are also highlighted in the table.

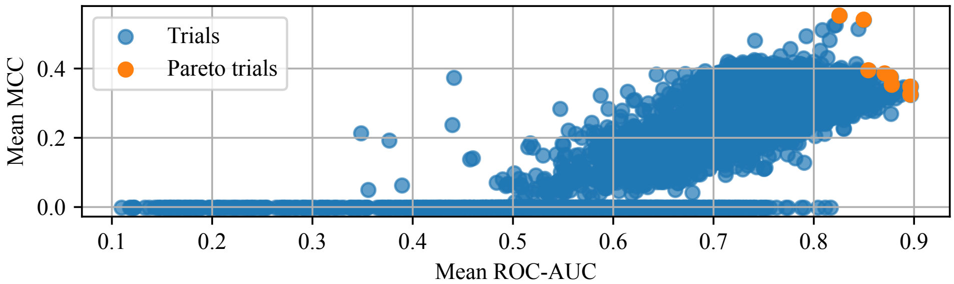

Figure 3. Cross-validation metrics of the models trained during NSGA-II parameter optimization.

Table 1. Trials in the Pareto front and objective metrics. The highest objective metrics achieved are highlighted in bold.

Trial	Mean ROC-AUC	Mean MCC
1205	0.878	0.353
2070	0.850	0.541
2423	0.876	0.376
3313	0.896	0.348
3825	0.854	0.395
3941	**0.896**	0.325
4366	0.826	**0.553**
4870	0.871	0.386

The relevance of parameter choice was evaluated based on the optimization trials. By means of graphical analysis, the parameters with the highest impact were the pre-processing ones (decimation factor, length of the cropped sequence, and choice of input measurement), followed by the learning rate, the size of the hidden layers of the LSTM, and the number of stacked LSTM layers. The analysis for these parameters, presented in Figure 4, indicated that the best results were obtained with mid-range decimation factors (65 and 125) and subsequences shorter than 200 samples, with lateral vibration providing the most informative signal. Regarding model parameters, learning rates around 10^{-3} and smaller LSTM hidden sizes yielded better performance, while deeper architectures showed no consistent advantage.

In order to evaluate the effectiveness of the proposed modeling and estimation technique, models trained using the parameter combinations from the Pareto trials and all labeled data were applied to label the previously unlabeled running-in phase of the tests. The resulting probabilities of steady-state for each unit and model are shown in Figure 5. Since the estimation results are directly linked to the representations extracted from the time series using the LSTM model, a bidimensional visualization of the higher-dimensional latent space was generated using uniform manifold approximation and projection (UMAP) for dimensionality reduction, and is shown in Figure 6. Both the probability estimation and the UMAP representation were used complementarily in the evaluation of the results.

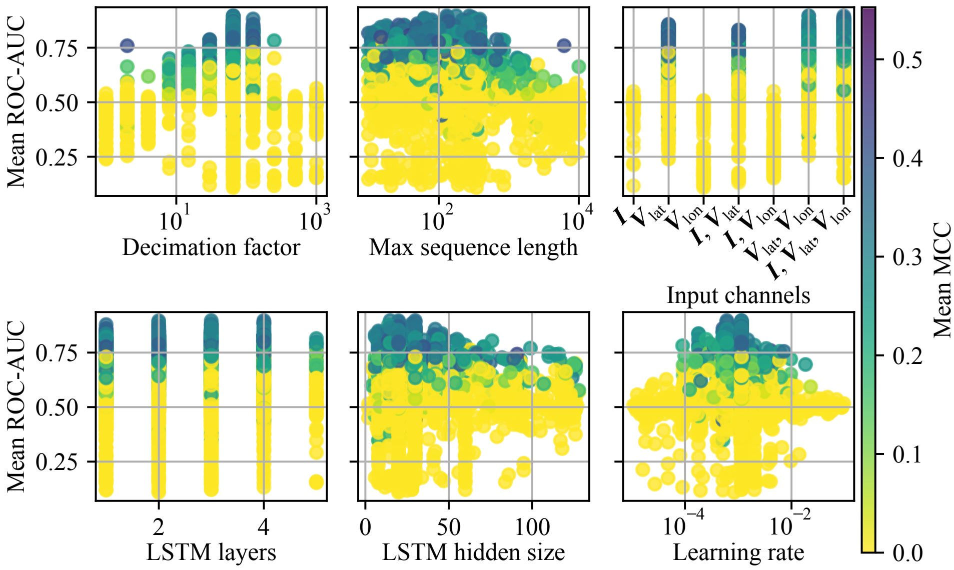

Figure 4. Parameter combinations with the highest impact in the cross-validation metrics.

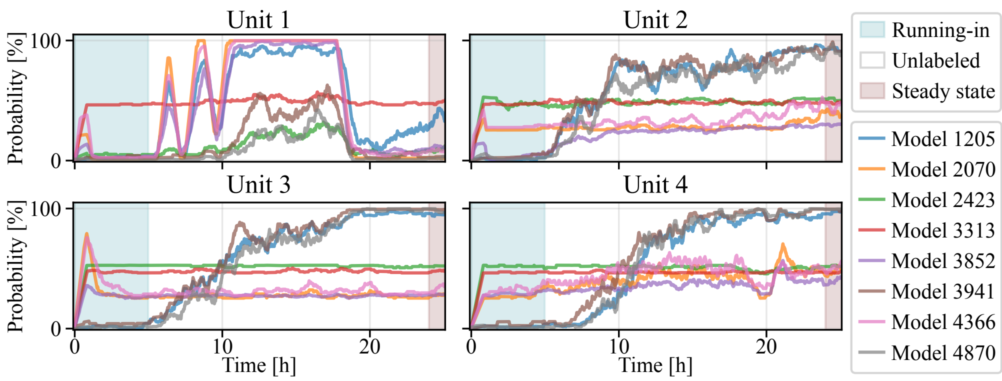

Figure 5. Estimated running-in probability for the evaluated compressor units.

In general, units 2, 3, and 4 presented similar results for each model, with results varying across models. Models 1205, 3941, and 4870 all presented a progression consistent with the running-in phenomenon (starting with a steady-state probability close to 0 %, then rising up to near 100 % along the first 24 h of the test). The latent space representation of these models was also the one that presented the clearest division between running-in and steady-state data: Model 1205 exhibited a radial-like distribution with running-in data at the center and steady-state data closer to the edges; Model 3941 exhibited a strip-like division with running-in data on one of the ends of the strips; and Model 4870 exhibited an elongated distribution with running-in data closer to the lower end of the shape.

The behavior of Unit 1, on the other hand, was significantly different from that observed in other units, with no model able to correctly estimate the steady-state at the end of the test. The difference between units is evident in the UMAP representation, as all models isolated Unit 1 from the others in the latent space, suggesting that this unit might be an outlier in terms of running-in behavior.

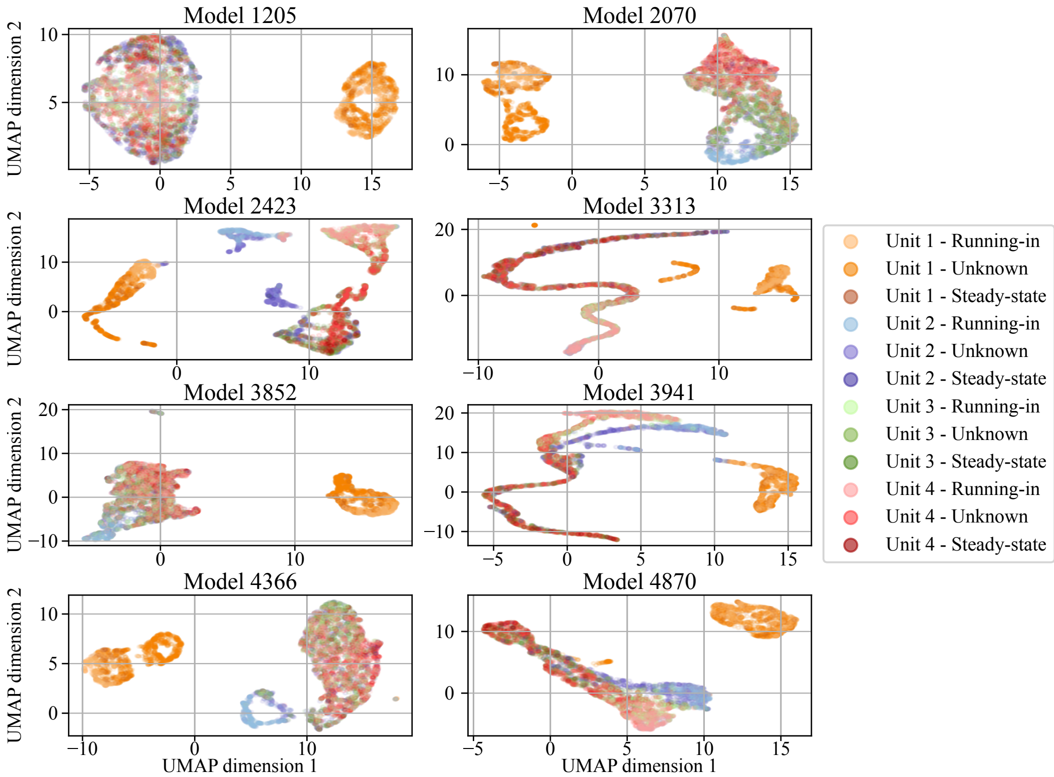

Figure 6. Latent space visualization with UMAP dimensionality reduction.

4. Conclusions

This work proposes the use of long short-term memory networks as a data-driven feature extraction approach for the analysis of the running-in phase of hermetic reciprocating compressors. By treating each high-frequency measurement window as an independent time series, the proposed framework focuses on extracting temporal features that characterize short-time dynamics without relying on handcrafted indicators or explicit assumptions about the duration of the running-in process.

Experimental results obtained from controlled compressor tests indicate that the LSTM-based representations capture consistent temporal signatures associated with running-in behavior across multiple units. The probabilistic estimation framework enabled a continuous characterization of the transition toward steady-state operation and highlighted differences between compressor units, including the identification of one unit with outlier behavior in the latent space.

The results suggest that recurrent neural networks can serve as effective feature extractors for gradual operational transients in rotating machinery, providing a viable alternative to the empirically defined running-in durations currently used in industrial practice. Future work will focus on extending the evaluation to a larger population of compressor units and on investigating the interpretability of the learned temporal features in relation to the physical mechanisms associated with the running-in phenomenon, as well as the integration with real-time monitoring and comparisons with alternative recurrent architectures or attention-based models.

Author Contributions: Conceptualization, T.R.v.C. and G.T.; methodology, T.R.v.C. and G.T.; software, G.T.; validation, G.T. and R.C.C.F.; formal analysis, R.C.C.F.; investigation, G.T.; resources, R.C.C.F.; data curation, G.T.; writing—original draft preparation, T.R.v.C. and G.T.; writing—review and editing, R.C.C.F.; visualization, G.T.; supervision, R.C.C.F.; project administration, R.C.C.F.; funding acquisition, R.C.C.F. All authors have read and agreed to the published version of the manuscript.

Funding: This study has been financially supported in part by the Human Resources Training Program of the Brazilian National Agency for Petroleum, Natural Gas and Biofuels—PRH-ANP, managed by the São Paulo Research Foundation (FAPESP), Brasil, Process number 2025/03789-1, in part by the Brazilian National Council for Scientific and Technological Development (CNPq) under Grant 310390/2026-5, and in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior-Brasil (CAPES)—Finance Code 001.

Data Availability Statement: The data that support the findings of this study are not openly available due to reasons of sensitivity and are available from the corresponding author upon reasonable request.

Acknowledgments: The authors acknowledge Nidec Global Appliance for the compressors used in the experiments.

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

References

1. Fawaz, H.I.; Forestier, G.; Weber, J.; Idoumghar, L.; Muller, P.A. Deep learning for time series classification: A review. *Data Min. Knowl. Discov.* **2019**, *33*, 917–963. [CrossRef]
2. Khan, S.; Yairi, T. A review on the application of deep learning in system health management. *Mech. Syst. Signal Process.* **2018**, *107*, 241–265. [CrossRef]
3. Lei, Y.; Li, N.; Guo, L.; Li, N.; Yan, T.; Lin, J. Machinery health prognostics: A systematic review from data acquisition to RUL prediction. *Mech. Syst. Signal Process.* **2018**, *104*, 799–834. [CrossRef]
4. Hennig, M.; Grafinger, M.; Hofmann, R.; Gerhard, D.; Dumss, S.; Rosenberger, P. Introduction of a time series machine learning methodology for the application in a production system. *Adv. Eng. Inform.* **2021**, *47*, 101197. [CrossRef]
5. Blau, P.J. On the nature of running-in. *Tribol. Int.* **2005**, *38*, 1007–1012. [CrossRef]
6. Khonsari, M.; Ghatrehsamani, S.; Akbarzadeh, S. On the running-in nature of metallic tribo-components: A review. *Wear* **2021**, *474–475*, 203871. [CrossRef]
7. *ISO 917:1989*; Testing of Refrigerant Compressors. International Organization for Standardization: Geneva, Switzerland, 1989.
8. ASHRAE. *ANSI/ASHRAE 23: Methods of Testing for Rating Positive Displacement Refrigerant Compressors and Condensing Units*; ASHRAE: Atlanta, GA, USA, 2005.
9. Thaler, G.; de S. Nascimento, A.S.; Pacheco, A.L.; Flesch, R.C. Unsupervised-ensemble-based method for automatic running-in information extraction in reciprocating compressors. *Adv. Eng. Inform.* **2024**, *62*, 102841. [CrossRef]
10. Martin, H. Vibration Analysis. In *ASM Handbook*; ASM International: Materials Park, OH, USA, 1992; Volume 18, pp. 555–568.
11. Eissenberg, D.; Haynes, H. Motor-Current Signature Analysis. In *ASM Handbook*; ASM International: Materials Park, OH, USA, 1992; Volume 18, pp. 594–605.
12. Soedel, W. *Sound and Vibrations of Positive Displacement Compressors*, 1st ed.; CRC Press: Boca Raton, FL, USA, 2007; p. 342.
13. Lv, Q.; Yu, X.; Ma, H.; Ye, J.; Wu, W.; Wang, X. Applications of machine learning to reciprocating compressor fault diagnosis: A review. *Processes* **2021**, *9*, 909. [CrossRef]
14. Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Comput.* **1997**, *9*, 1735–1780. [CrossRef] [PubMed]
15. Yang, X.; Wan, C.; Zhang, T.; Xiong, Z. Feature extraction of sequence data based on LSTM and its application to fault diagnosis of industrial process. In Proceedings of the 2022 IEEE 11th Data Driven Control and Learning Systems Conference (DDCLS), Chengdu, China, 3–5 August 2022; pp. 693–698. [CrossRef]
16. Gangsar, P.; Bajpei, A.R.; Porwal, R. A review on deep learning based condition monitoring and fault diagnosis of rotating machinery. *Noise Vib. Worldw.* **2022**, *53*, 550–578. [CrossRef]
17. Deb, K.; Pratap, A.; Agarwal, S.; Meyarivan, T. A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans. Evol. Comput.* **2002**, *6*, 182–197. [CrossRef]
18. Torky, A.A.; Ohno, S. Deep learning techniques for predicting nonlinear multi-component seismic responses of structural buildings. *Comput. Struct.* **2021**, *252*, 106570. [CrossRef]
19. Wiesner, D.; Schirmeier, F. Performance and Computation Time Gains Caused by Sampling Rate Reduction in Time Series Deep Anomaly Detection. In *Computer Aided Systems Theory–EUROCAST 2024. EUROCAST 2024*; Lecture Notes in Computer Science; Springer Nature: Cham, Switzerland, 2025; Volume 15174, pp. 176–186. [CrossRef]
20. Stoica, P.; Babu, P. Pearson–Matthews Correlation Coefficients for Binary and Multinary Classification. *Signal Process.* **2024**, *222*, 109511. [CrossRef]
21. Itaya, Y.; Tamura, J.; Hayashi, K.; Yamamoto, K. Asymptotic Properties of Matthews Correlation Coefficient. *Stat. Med.* **2025**, *44*, e10303. [CrossRef] [PubMed]

22. Chicco, D.; Jurman, G. The Matthews Correlation Coefficient (MCC) Should Replace the ROC AUC as the Standard Metric for Assessing Binary Classification. *BioData Min.* **2023**, *16*, 4. [CrossRef] [PubMed]
23. Raftopoulos, G.; Kostopoulos, G.; Davrazos, G.; Panagiotakopoulos, T.; Kotsiantis, S.; Kameas, A. Comparative Analysis of Self-Labeled Algorithms for Predicting MOOC Dropout: A Case Study. *Appl. Sci.* **2025**, *15*, 12025. [CrossRef]
24. Naser, M.Z.; Alavi, A.H. Error Metrics and Performance Fitness Indicators for Artificial Intelligence and Machine Learning in Engineering and Sciences. *Archit. Struct. Constr.* **2023**, *3*, 499–517. [CrossRef]