Temporal Conditional Normalizing Flows for Data Augmentation in Remaining Useful Life Prediction under Data Scarcity

Guillaume Prevost[1], Esteban Cabanillas[1], Jérôme Boutet[1], Cornel Ioana[2]

[1] Univ. Grenoble Alpes, CEA, Leti, F-38000 Grenoble, France

[2] Gipsa-Lab, Univ. Grenoble-Alpes, Saint-Martin d'Hères, France

guillaume.prevost@cea.fr

Résumé

La prédiction de la durée de vie résiduelle (RUL) des roulements souffre d'un manque de données industrielles, limitant la généralisation des modèles supervisés. Nous proposons un modèle TCNF (Temporal Conditional Normalizing Flow), un modèle génératif conditionné sur la RUL et un contexte temporel encodé par GRU, capable de générer des trajectoires de dégradation complètes et cohérentes. Une couche de normalisation affine apprise est intégrée directement dans le flot, garantissant bijectivité et calcul exact du log-déterminant. Sur le jeu de données XJTU-SY, TCNF améliore le RMSE par rapport à la référence sans augmentation et surpasse les approches comparatives GAN et VAE.

Mots-clés

Normalizing flows, augmentation de données, durée de vie résiduelle.

Abstract

Predicting the remaining useful life (RUL) of bearings suffers from a lack of industrial data, limiting the generalization of supervised models. We propose a temporal conditional normalizing flow model (TCNF), a generative model conditioned on RUL and a temporal context encoded by GRU, capable of generating complete and consistent degradation trajectories. An affine normalization layer is integrated directly into the flow, ensuring bijectivity and exact log-determinant computation. On the XJTU-SY dataset, TCNF improves the RMSE compared to the baseline without augmentation and outperforms the comparative GAN and VAE approaches.

Keywords

Normalizing flows, data augmentation, remaining useful life.

1 Introduction

The predictive maintenance (PdM) field has become a major strategic lever for industry, enabling reduced maintenance costs, improved equipment availability and prevention of critical failures, with significant economic gains [1][2]. At the core of PdM, Remaining Useful Life (RUL) estimation aims to predict the time remaining before system failure from sensor measurements such as vibrations, currents or temperatures. In rotating mechanical systems, and in particular bearings, vibration signals are widely exploited due to their high sensitivity to degradation mechanisms [3] [4].

Despite recent advances in deep learning methods for RUL estimation [5][6], their industrial deployment remains severely limited by the scarcity of relevant data. Run-to-failure data, required to model the complete evolution of degradation up to failure, are costly to acquire, time-consuming to collect, and rarely available in large quantities [7]. This constraint is even more pronounced in embedded or real industrial settings, where equipment is designed precisely to avoid failure situations. As a result, datasets used for RUL model training are typically small, imbalanced and heterogeneous, leading to overfitting and limited generalization.

To address this data scarcity, several recent works have explored manual data augmentation techniques [8][9], or generative model-based approaches such as Generative Adversarial Networks (GAN) [10][11] or Variational Autoencoders (VAE) [12][13]. Although these approaches have shown promising potential for enriching training sets, they exhibit important limitations in the context of predictive maintenance. GANs frequently suffer from training instability and mode collapse, while VAEs rely on likelihood approximations that can degrade the statistical fidelity of generated data. Moreover, most of these approaches generate samples independently, without guaranteeing the temporal coherence of degradation trajectories or explicit conditioning on health state, limiting their application to fault classification [14][15].

Yet in RUL applications, system degradation is inherently a temporal and progressive phenomenon. Many works have shown that RUL can be modelled according to a two-phase dynamics, comprising an initial healthy operating phase followed by an accelerated degradation phase leading to failure [16]. Generating realistic synthetic data should therefore respect not only the marginal distributions of measured features, but also their temporal evolution conditioned on RUL. Few existing works explicitly address this problem within a rigorous probabilistic framework.

In this context, normalizing flow models [17][18] appear as a particularly well-suited alternative for data generation in predictive maintenance. These generative models rely on bijective and differentiable transformations that enable exact probability density estimation through likelihood maximization. Unlike GANs and VAEs, normalizing flows offer increased training stability, clear probabilistic interpretability, and coherent sample generation through model inversion. However, their application to the generation of temporally conditioned degradation trajectories remains largely unexplored in the literature.

In this paper, we propose a novel data augmentation approach for RUL estimation based on a Temporal Conditional Normalizing Flow, specifically designed for predictive maintenance scenarios under limited data constraints. The proposed model incorporates explicit RUL conditioning as well as temporal context encoding through a recurrent network, enabling the generation of complete, temporally coherent degradation trajectories. To guarantee model bijectivity and compatibility with normalizing flows, a learned, invertible affine normalization layer is introduced as a replacement for conventional normalization methods. The generated synthetic data are then used to enrich the training set of a RUL prediction model.

The main contributions of this paper are as follows:

- the proposal of a RUL-conditioned Normalizing Flow, capable of modelling the distribution of vibration features throughout the entire lifetime of a bearing,
- the integration of a recurrent temporal context enabling the generation of coherent and realistic degradation trajectories,
- the introduction of a learned, bijective and invertible affine normalization layer, specifically adapted to the constraints of flow-based models.

The remainder of this paper is organized as follows. Section 2 describes the proposed approach in detail. Section 3 presents the experimental protocol, while Sections 4 and 5 discuss the obtained results. Finally, Section 6 concludes the paper and outlines future research directions.

2 Proposed methodology

2.1 Overview of the framework

Our approach relies on a three-stage pipeline. In the first stage, d features are extracted from raw vibration signals and concatenated into a feature vector at each time step. These vectors, from the start of system life to the end of life, form degradation trajectories on which the generative model is trained. In the second stage, the model is used to generate new complete synthetic trajectories, which are added to the training set of a RUL prediction model. Fig 1. Shows the overall pipeline of the proposed approach. A RUL prediction regressive model is then trained on the augmented set.

The objective of the generative model is to learn the conditional distribution of the feature vectors observed at each time step, given the current RUL value and the recent observation history. Formally, we seek to model:

$$p(x_t | y_t, x_{\{t-1:t-k\}})$$

where $x_t \in \mathbb{R}^d$ is the feature vector at time t, $y_t \in [0,1]$ is the normalized RUL, and $x_{\{t-\mathrm{k}:t-1\}}$ represents the k previous observations. A complete degradation trajectory is then generated by sequentially sampling from this distribution, from $y = 1.0$ down to $y = 0.0$. This work proposes to use a normalizing flow model conditioned on RUL and temporal context to generate complete degradation trajectories.

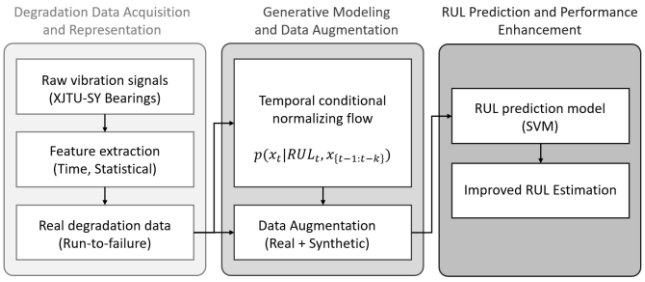

Figure 1. Overall pipeline of the proposed method. The pipeline consists of three stages: (1) degradation data acquisition and representation, (2) generative modelling and data augmentation, (3) RUL prediction.

2.2 Extracted features

The features extracted from vibration signals are chosen to capture different aspects of degradation. We retain classical time-domain descriptors from the literature: root mean square (RMS), zero-crossing rate (ZCR), crest factor, skewness, peak-to-peak amplitude, mean and kurtosis. These descriptors are widely used in the bearing health monitoring literature and enable the capture of phenomena of different natures: overall signal energy, impulsiveness, and non-stationarities [19].

2.3 RUL modelisation

In line with common practice in the predictive bearing maintenance literature, RUL is modelled as a two-phase function: a healthy phase followed by a degradation phase [16]. This model, often referred to as a piecewise linear RUL model [20], provides a more accurate representation of the physical dynamics of degradation by explicitly distinguishing the nominal operating phase from the active degradation phase [21].

Formally, for a bearing with total lifetime T, the normalized RUL at time t is defined by:

$$y_t = \begin{cases} 1, & t < t_{FPT} \\ 1 - \dfrac{t - t_{FPT}}{T - t_{FPT}}, & t \geq t_{FPT} \end{cases} \quad (1)$$

where t_{FPT} (First Predictable Time) denotes the instant from which degradation becomes measurable. This formulation has the advantage of limiting the influence of noise during the healthy phase and of introducing structural constraints (monotonicity and boundedness between 0 and 1) consistent with the physics of the system [22]. In this study, the t_{FPT} values are set in accordance with the proposals of Yin et al. [23].

2.4 Temporal Conditional Normalizing Flow

2.4.1 General structure

The Temporal Conditional Normalizing Flow (TCNF) consists of a succession of conditioned affine coupling layers, preceded by the learned normalization layer. The core idea of normalizing flows is to learn a bijective and differentiable mapping between a complex data distribution and a simple base distribution, here a standard Gaussian, such that both exact sampling and exact likelihood evaluation are tractable. The model transforms a data vector x into a latent vector z by:

$$z = f_K \circ f_{K-1} \circ \dots \circ f_1 \circ Norm(x)$$

where each f_i is an affine coupling layer. The exact log-likelihood is computed via the change-of-variables theorem:

$$log\ p(x) = log\ p_z(z) + \sum_{i=1}^{K} log\ |det\ J_{f_i}| + log\ |det\ J_{Norm}| \quad (2)$$

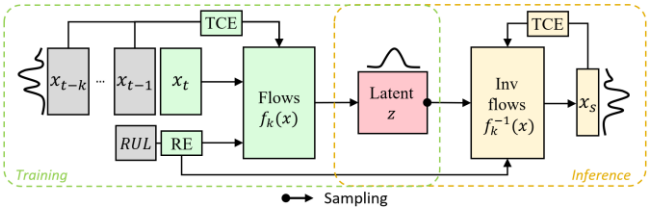

Figure 2. Architecture of the TCNF during training and inference phases. The TCE (Temporal Context Encoder) encodes the sliding history $x_{t-k}, \dots, x_t$ via a gated recurrent unit (GRU), while the RUL Encoder (RE) projects the current RUL value into an embedding space.

2.4.2 Learned affine normalization

Adequate data preprocessing is crucial for training normalizing flows. A traditional normalization scheme has the drawback of being a transformation external to the model, non-differentiable with respect to its parameters, and whose statistics must be computed and stored separately. We instead propose a learned affine normalization layer, integrated directly as the first layer of the flow.

Let $x \in \mathbb{R}^d$ be an input vector. The transformation is defined by:

$$z = \frac{(x - \mu)}{\sigma} \quad (3)$$

where μ and $\sigma = exp(log\_\sigma)$ are learnable parameters initialized from the empirical statistics of the training set. The constraint $\sigma > 0$ is guaranteed by the log_σ parameterization. The associated log-Jacobian determinant is:

$$log\ |det\ J| = -\sum_{i=1}^{d} \log\ (\sigma_i) \quad (4)$$

This formulation offers several decisive advantages over standard normalization: bijectivity is guaranteed by construction, the behavior is identical during training and inference, and the log-determinant is computable explicitly and exactly, which is a fundamental requirement of normalizing flows. Furthermore, the parameters are initialized from training set statistics and can be refined through backpropagation during training. Note that a version without log-determinant computation is also available to normalize the history before passing it to the temporal encoder, avoiding an unnecessary computation in this context.

2.4.3 Affine coupling layer

Each coupling layer splits the input vector into two parts via a binary mask that alternates between layers. The masked part x_1 is passed unchanged, while the unmasked part x_2 is transformed by an affine transformation whose parameters depend on both x_1 and the condition c:

$$x_2' = x_2 \times exp\big(s(x_1, c)\big) + t(x_1, c) \quad (5)$$

where s and t are neural networks with two hidden layers and ReLU activations, and c is the condition vector integrating the RUL embedding and the temporal context. The inverse transformation is trivial and the log-determinant equals the sum over the unmasked dimensions of $s(x_1, c)$ guaranteeing computation in linear time.

2.4.4 RUL Encoding

The scalar RUL value $y_t \in [0, 1]$ is projected into a higher-dimensional space. This allows the model to learn a non-linear representation of the RUL condition, more expressive than a simple concatenation of the raw scalar value.

2.4.5 Temporal context encoder (TCE)

To capture degradation dynamics and ensure the temporal coherence of generated trajectories, we introduce a Temporal Context Encoder (TCE) based on a GRU network. At each time step t during generation, the k most recent observations are normalized (via the learned affine normalization without log-det) and passed to the GRU, whose final hidden state is projected into a context vector $c_t \in \mathbb{R}^{d_{ctx}}$:

$$c_t = MLP(GRU(Norm(x_{t-k}, \dots, x_{t-1})))$$

This context is concatenated with the encoded RUL vector to form the complete condition for each coupling layer. In the absence of history (beginning of a trajectory), the context is set to zero, allowing the model to generate a coherent first point from the learned distribution.

2.5 Training

The model is trained by maximizing the exact log-likelihood (or equivalently minimizing the NLL). The training objective writes:

$$\mathcal{L} = -\left(\frac{1}{N}\right)\sum_{t=1}^{N} log\ p(x_t\ |y_t, x_{\{t-k:t-1\}}) \quad (6)$$

Temporal sequences are constructed by extracting sliding windows of length k over each bearing trajectory. To strengthen model robustness and enable trajectory generation in the absence of temporal context (e.g. at the beginning of a trajectory), a fraction of training examples is processed without temporal context (context set to zero). In our experiments, this ratio is set to 0.2.

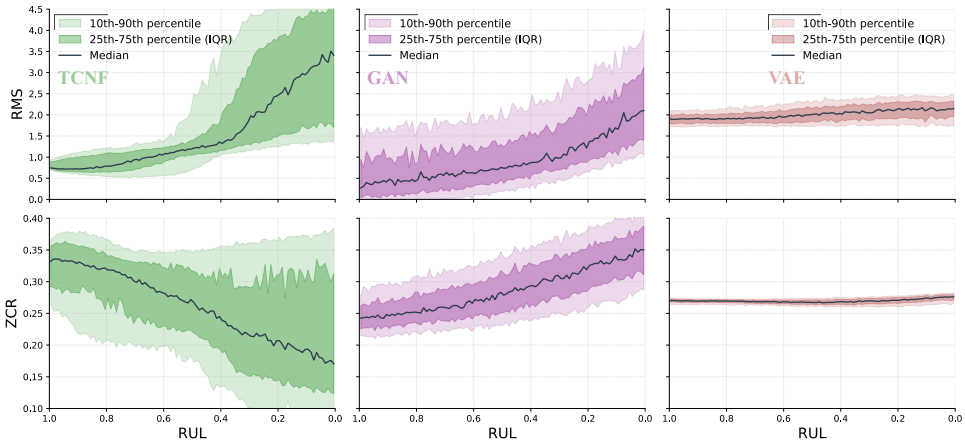

Figure 4. Mean generated trajectories for RMS and ZCR by TCNF (green), GAN (purple) and VAE (brown). Solid curves represent the median over 200 generated trajectories. Shaded areas indicate the 25–75% and 10–90% intervals.

3 Experimental validation

3.1 XJTU-SY dataset

The method is validated on the XJTU-SY experimental dataset, developed by Xi'an Jiaotong University in collaboration with Changxing Sumyoung Technology. This dataset provides 15 run-to-failure bearing experiments under three different load and rotation speed conditions (5 experiments per condition), capturing a wide diversity of lifetimes and fault types (inner race, outer race, cage and ball defects).

Vibration signals are sampled at 25.6 kHz, with 32,768 samples acquired every minute. A 5-fold cross-validation protocol is employed to assess model robustness and generalization ability. At each iteration, bearings are split into training and test sets with no overlap. Hyperparameters are kept identical across all folds to ensure fair comparison.

3.2 Comparative approaches

We compare TCNF against two generative baselines representative of the state of the art in data augmentation: a Generative Adversarial Network (GAN) and a Variational Autoencoder (VAE). Both models are equally conditioned on RUL and trained on the same dataset. The configuration without augmentation (training on real data only) serves as the absolute reference (Baseline).

4 Synthetic data quality

The quality of synthetic data generated by TCNF is assessed along two complementary axes: (i) analysis of the statistical consistency of conditional distributions, and (ii) analysis of the temporal coherence of complete generated trajectories.

4.1 Conditional distribution analysis

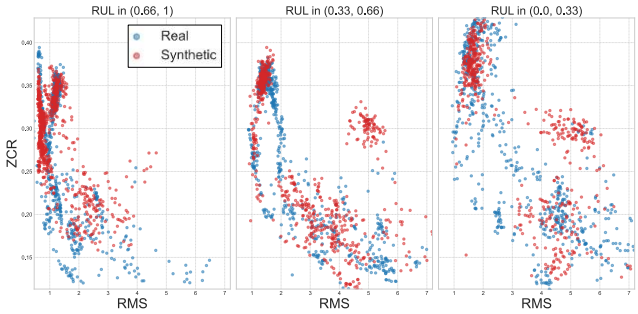

Figure 3. Joint RMS–ZCR distribution for three RUL intervals (high, intermediate, low). Blue points correspond to real training data, red points to synthetic data generated by TCNF.

Fig. 3 illustrates the joint distribution of the RMS and ZCR features for three RUL intervals (high, medium and low). Real data are shown in blue, while synthetic data generated by TCNF are shown in red.

The model faithfully reproduces the multivariate and non-linear structure of the real distribution. To quantify this agreement, we compute the Maximum Mean Discrepancy (MMD) between real and synthetic distributions within each RUL bin, averaged over 10 repetitions to account for generation stochasticity. TCNF achieves a mean MMD of 0.15 (± 0.06), outperforming both GAN (0.22 ± 0.01) and VAE (0.42 ± 0.01). The higher variance of TCNF's MMD across repetitions reflects its broader generative diversity, consistent with the wider confidence intervals observed in trajectory analysis. As RUL decreases, the distribution spreads and deforms, reflecting the increase in vibration variability and the emergence of impulsive behaviors typical of bearing degradation.

TCNF correctly captures:

- the progressive drift of distribution centroids,
- the increase in dispersion towards end of life,
- non-Gaussian structures and inter-feature correlations.

This ability to model a complex distribution conditioned on time confirms the benefit of joint conditioning on both RUL and temporal context.

4.2 Temporal coherence of generated trajectories

Fig. 4 compares the synthetic trajectories generated by TCNF, GAN and VAE. For each model, 200 complete trajectories are generated, and the median along with interquartile (25–75%) and extended (10–90%) intervals are shown for the RMS and ZCR features.

TCNF is distinguished by:

- realistic non-linear dynamics,
- a relatively stable healthy phase and a progressive transition towards an accelerated degradation phase,
- and increasing variability towards end of life.

Conversely, GAN produces smoother and more monotonic trajectories. Inter-trajectory variability is lower and the transition between healthy and degraded phases appears less coherent. VAE generates trajectories close to zero, with strongly reduced variability.

These observations confirm that the integration of the Temporal Context Encoder (TCE) enables TCNF to capture realistic degradation dynamics while preserving inter-trajectory diversity.

4.3 Contitioned extrapolation

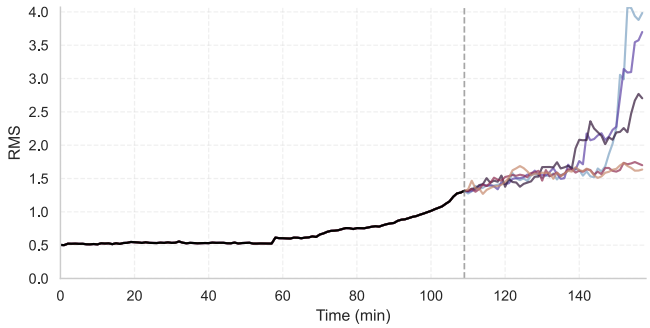

Figure 5. Example of conditioned extrapolation. The left part corresponds to a real partial trajectory (RMS). Colored curves represent five synthetic continuations generated by TCNF.

An important advantage of the proposed model lies in the ability to initialize the TCE from a real partial history. The model can thereby extrapolate an existing trajectory by generating its continuation conditioned on the current state.

Figure 5 presents an extrapolation example from a real trajectory (black line, left part). Five synthetic continuations are generated. It can be observed that:

- the extrapolated trajectories remain consistent with the initial dynamics,
- dispersion naturally increases during the degradation phase,
- and some trajectories exhibit a marked acceleration, reflecting the intrinsic uncertainty of end-of-life behavior.

This capability opens the way to targeted augmentation strategies, in particular for enriching data in the terminal phase (low RUL), which is often under-represented in industrial datasets.

5 Impact on RUL prediction

To evaluate the contribution of synthetic data, we train two regressive models to predict RUL with and without synthetic data.

Support Vector Machine (SVM). The SVM is widely used in the literature for RUL prediction and fault classification [24][25]. The input vector is augmented with an Exponentially Weighted Moving Average (EWMA) at two temporal scales, informing the model about the trend of each feature's evolution at different time scales.

Recurrent Neural Network (RNN). A recurrent neural network using a gated recurrent unit GRU architecture, with greater non-linear modelling capacity than the SVM, and also widely used in predictive maintenance for time series analysis [26][27].

Performance is evaluated according to three complementary criteria. RMSE measures the overall quadratic error:

$$RMSE = \sqrt{\frac{1}{T}\sum_{t=1}^{T}(\hat{y}_t - y_t)^2} \tag{7}$$

Mean absolute error (MAE) measures the error in absolute value:

$$MAE = \frac{1}{T}\sum_{t=1}^{T}|\hat{y}_t - y_t| \tag{8}$$

The NASA score asymmetrically penalizes early predictions (underestimation of remaining RUL, risk of unwarranted maintenance) and late predictions (overestimation, risk of unanticipated failure), the latter being more critical in an industrial context:

$$Score = \sum_{t=1}^{T}\begin{cases}\exp\left(-\frac{\hat{y}_t - y_t}{13}\right) - 1, & \hat{y}_t < y_t \\ \exp\left(\frac{\hat{y}_t - y_t}{10}\right) - 1, & \hat{y}_t \geq y_t\end{cases} \tag{9}$$

To limit the impact of the stochastic nature inherent to generation, each experiment is repeated 10 times and reported results correspond to the means over these 10 repetitions. In each configuration, 7 synthetic trajectories are generated (corresponding to approximately 70% additional synthetic data). Table 1 presents the RUL prediction results for the four evaluated configurations.

Table 1. SVM RUL prediction results.

	RMSE	MAE	Nasa-score
Baseline	0.222	0.172	0.015
GAN	0.219	0.168	0.015
VAE	0.232	0.188	0.016
TCNF	**0.205**	**0.157**	**0.014**

Table 2. RNN RUL prediction results.

	RMSE	MAE	Nasa-score
Baseline	0.179	0.124	0.022
GAN	0.167	0.115	0.022
VAE	0.172	0.115	0.024
TCNF	**0.159**	**0.099**	**0.020**

TCNF improves all metrics for both prediction models. For the SVM, RMSE decreases by 7.7%, MAE by 8.7% and the NASA score by 6.7%. Gains are even more pronounced for the RNN, with RMSE dropping from 0.179 to 0.159 (−11.2%). These improvements, consistent across all cross-validation folds, indicate that synthetic trajectories bring genuinely useful information, regardless of the capacity of the downstream model.

The comparison between baselines is instructive. GAN provides modest but systematic gains on both models. VAE, on the other hand, slightly improves RNN performance but degrades SVM performance relative to the no-augmentation baseline, suggesting that its variational regularization produces trajectories insufficiently representative of the real variability of degradation.

The fact that TCNF is the only model to consistently improve the NASA score — the most demanding metric due to its asymmetric nature — is relevant: it suggests that the temporal coherence ensured by the TCE leads to better representation of end-of-life dynamics, where prediction errors are most costly from an industrial standpoint.

A further advantage of the approach lies in the ability to generate trajectories specifically targeting the degradation phase ($RUL < 1$), allowing rebalancing of the natural imbalance between the healthy phase (often the majority) and the degraded phase. By setting a short t_{FPT} during generation, it is possible to produce synthetic trajectories with an extended degradation phase, particularly enriching the representation of the most critical health states.

6 Conclusion

This paper introduced the Temporal Conditional Normalizing Flow (TCNF), a generative approach dedicated to data augmentation for remaining useful life prediction in industrial data scarcity settings. By combining a conditional normalizing flow, explicit RUL encoding and a GRU-modelled temporal context, the model generates complete, temporally coherent and statistically faithful degradation trajectories.

Results obtained on the XJTU-SY dataset show that synthetic data produced by TCNF systematically improve prediction performance (RMSE, MAE and NASA score), in contrast to the GAN and VAE-based comparative approaches. These gains confirm the value of exact probabilistic modelling and explicit temporal conditioning for degradation data generation.

Beyond quantitative performance, the model enables targeted generation of critical degradation phases and conditioned extrapolation of partial trajectories, opening promising avenues for more refined augmentation strategies in predictive maintenance.

This work highlights the potential of temporal conditional normalizing flows as a robust and relevant tool for improving the generalization of RUL prediction models in constrained industrial settings.

7 References

[1] J. Guo, Z. Li, and M. Li, ‘A Review on Prognostics Methods for Engineering Systems’, *IEEE Trans. Rel.*, vol. 69, no. 3, pp. 1110–1129, Sep. 2020, doi: 10.1109/TR.2019.2957965.

[2] M. Haarman, M. Mulders, and C. Vassiliadis, ‘Predictive maintenance 4.0: predict the unpredictable.’, *PwC and Mainnovation, 4*, 2017.

[3] A. Althubaiti, F. Elasha, and J. A. Teixeira, ‘Fault diagnosis and health management of bearings in rotating equipment based on vibration analysis – a review’, *J. vibroeng.*, vol. 24, no. 1, pp. 46–74, Feb. 2022, doi: 10.21595/jve.2021.22100.

[4] D. Jung, Z. Zhang, and M. Winslett, ‘Vibration Analysis for IoT Enabled Predictive Maintenance’, in *2017 IEEE 33rd International Conference on Data Engineering (ICDE)*, San Diego, CA, USA: IEEE, Apr. 2017, pp. 1271–1282. doi: 10.1109/ICDE.2017.170.

[5] Y. Lei, N. Li, L. Guo, N. Li, T. Yan, and J. Lin, ‘Machinery health prognostics: A systematic review from data acquisition to RUL prediction’, *Mechanical Systems and Signal Processing*, vol. 104, pp. 799–834, May 2018, doi: 10.1016/j.ymssp.2017.11.016.

[6] S. Khan and T. Yairi, ‘A review on the application of deep learning in system health management’, *Mechanical Systems and Signal Processing*, vol. 107, pp. 241–265, Jul. 2018, doi: 10.1016/j.ymssp.2017.11.024.

[7] J. Zhang, D. Zhang, M. Yang, X. Xu, W. Liu, and C. Wen, ‘Fault Diagnosis for Rotating Machinery with Scarce Labeled Samples: A Deep CNN Method Based on Knowledge-Transferring from Shallow Models’, in *2018 International Conference on Control, Automation and Information Sciences (ICCAIS)*, Hangzhou: IEEE, Oct. 2018, pp. 482–487. doi: 10.1109/ICCAIS.2018.8570515.

[8] T. Peng, C. Shen, S. Sun, and D. Wang, ‘Fault Feature Extractor Based on Bootstrap Your Own Latent and Data Augmentation Algorithm for Unlabeled Vibration Signals’, *IEEE Trans. Ind. Electron.*, vol. 69, no. 9, pp. 9547–9555, Sep. 2022, doi: 10.1109/TIE.2021.3111567.

[9] M. Hu, C. Wang, C. Zhuang, and Y. Wang, ‘Bearing fault diagnosis method based on data augmentation and

MCNN-LSTM', in *2023 IEEE 6th Information Technology,Networking,Electronic and Automation Control Conference (ITNEC)*, Chongqing, China: IEEE, Feb. 2023, pp. 663–671. doi: 10.1109/ITNEC56291.2023.10082598.

[10] T. Koenig, L. Cadau, F. Wagner, and M. Kley, 'A generative adversarial network-based data augmentation approach with transient vibration data', *Procedia Computer Science*, vol. 225, pp. 1340–1349, 2023, doi: 10.1016/j.procs.2023.10.122.

[11] S. Shao, P. Wang, and R. Yan, 'Generative adversarial networks for data augmentation in machine fault diagnosis', *Computers in Industry*, vol. 106, pp. 85–93, Apr. 2019, doi: 10.1016/j.compind.2019.01.001.

[12] M. S. Rathore and S. P. Harsha, 'Non-linear Vibration Response Analysis of Rolling Bearing for Data Augmentation and Characterization', *J. Vib. Eng. Technol.*, vol. 11, no. 5, pp. 2109–2131, Jul. 2023, doi: 10.1007/s42417-022-00691-w.

[13] L. Wang, Q. Qu, Y. Wang, D. S.-H. Wong, and Y. Zheng, 'Data Augmentation Integrated with Feature-Enhanced Convolutional Neural Network for Imbalanced Fault Diagnosis in Rolling Bearings', in *2024 IEEE 13th Data Driven Control and Learning Systems Conference (DDCLS)*, Kaifeng, China: IEEE, May 2024, pp. 1204–1209. doi: 10.1109/DDCLS61622.2024.10606651.

[14] S. Sun, H. Ding, H. Huang, Z. Zhao, D. Wang, and W. Xu, 'A Novel Cross-Domain Data Augmentation and Bearing Fault Diagnosis Method Based on an Enhanced Generative Model', *IEEE Trans. Instrum. Meas.*, vol. 73, pp. 1–9, 2024, doi: 10.1109/TIM.2024.3390242.

[15] X. Yang, T. Ye, X. Yuan, W. Zhu, X. Mei, and F. Zhou, 'A Novel Data Augmentation Method Based on Denoising Diffusion Probabilistic Model for Fault Diagnosis Under Imbalanced Data', *IEEE Trans. Ind. Inf.*, vol. 20, no. 5, pp. 7820–7831, May 2024, doi: 10.1109/TII.2024.3366991.

[16] N. Li, Y. Lei, J. Lin, and S. X. Ding, 'An Improved Exponential Model for Predicting Remaining Useful Life of Rolling Element Bearings', *IEEE Trans. Ind. Electron.*, vol. 62, no. 12, pp. 7762–7773, Dec. 2015, doi: 10.1109/TIE.2015.2455055.

[17] D. Rezende and S. Mohamed, 'Variational Inference with Normalizing Flows', in *Proceedings of the 32nd International Conference on Machine Learning*, PMLR, Jun. 2015, pp. 1530–1538. Accessed: Feb. 25, 2026. [Online]. Available: https://proceedings.mlr.press/v37/rezende15.html

[18] M. Russell and P. Wang, 'Normalizing Flows for Intelligent Manufacturing', in *Volume 2: Manufacturing Equipment and Automation; Manufacturing Processes; Manufacturing Systems; Nano/Micro/Meso Manufacturing; Quality and Reliability*, New Brunswick, New Jersey, USA: American Society of Mechanical Engineers, Jun. 2023, p. V002T09A004. doi: 10.1115/MSEC2023-101281.

[19] Y. Wang, J. Zhao, C. Yang, D. Xu, and J. Ge, 'Remaining useful life prediction of rolling bearings based on Pearson correlation-KPCA multi-feature fusion', *Measurement*, vol. 201, p. 111572, Sep. 2022, doi: 10.1016/j.measurement.2022.111572.

[20] C. Yin, Y. Li, Y. Wang, and Y. Dong, 'Physics-guided degradation trajectory modeling for remaining useful life prediction of rolling bearings', *Mechanical Systems and Signal Processing*, vol. 224, p. 112192, Feb. 2025, doi: 10.1016/j.ymssp.2024.112192.

[21] G. Wang and J. Xiang, 'Remain useful life prediction of rolling bearings based on exponential model optimized by gradient method', *Measurement*, vol. 176, p. 109161, May 2021, doi: 10.1016/j.measurement.2021.109161.

[22] Y. Qin, C. Yuen, Y. Shao, B. Qin, and X. Li, 'Slow-Varying Dynamics-Assisted Temporal Capsule Network for Machinery Remaining Useful Life Estimation', *IEEE Trans. Cybern.*, vol. 53, no. 1, pp. 592–606, Jan. 2023, doi: 10.1109/TCYB.2022.3164683.

[23] C. Yin, Y. Li, Y. Wang, and Y. Dong, 'Physics-guided degradation trajectory modeling for remaining useful life prediction of rolling bearings', *Mechanical Systems and Signal Processing*, vol. 224, p. 112192, Feb. 2025, doi: 10.1016/j.ymssp.2024.112192.

[24] M. Yan, X. Wang, B. Wang, M. Chang, and I. Muhammad, 'Bearing remaining useful life prediction using support vector machine and hybrid degradation tracking model', *ISA Transactions*, vol. 98, pp. 471–482, Mar. 2020, doi: 10.1016/j.isatra.2019.08.058.

[25] M. Pandiyan and T. N. Babu, 'Systematic Review on Fault Diagnosis on Rolling-Element Bearing', *J. Vib. Eng. Technol.*, vol. 12, no. 7, pp. 8249–8283, Oct. 2024, doi: 10.1007/s42417-024-01358-4.

[26] J. Zhou, Y. Qin, D. Chen, F. Liu, and Q. Qian, 'Remaining useful life prediction of bearings by a new reinforced memory GRU network', *Advanced Engineering Informatics*, vol. 53, p. 101682, Aug. 2022, doi: 10.1016/j.aei.2022.101682.

[27] L. Song, T. Lin, Y. Jin, S. Zhao, Y. Li, and H. Wang, 'Advancements in bearing remaining useful life

prediction methods: a comprehensive review', *Meas. Sci. Technol.*, vol. 35, no. 9, p. 092003, Sep. 2024, doi: 10.1088/1361-6501/ad5223.