

Article

Not peer-reviewed version

A Two-Stage Deep Learning Framework for Uncertainty-Aware Forecasting and Conditional Process Optimization Toward Sustainable Smart Manufacturing

[Junhee Lee](#)[†], [Sajid Ali](#)[†], [Hail Jung](#)^{*}

Posted Date: 6 August 2025

doi: 10.20944/preprints202508.0377.v1

Keywords: Industry 4.0; artificial intelligence; uncertainty-aware forecasting; generative modeling; process control; sustainable manufacturing



Preprints.org is a free multidisciplinary platform providing preprint service that is dedicated to making early versions of research outputs permanently available and citable. Preprints posted at Preprints.org appear in Web of Science, Crossref, Google Scholar, Scilit, Europe PMC.

Copyright: This open access article is published under a Creative Commons CC BY 4.0 license, which permit the free download, distribution, and reuse, provided that the author and preprint are cited in any reuse.

Disclaimer/Publisher's Note: The statements, opinions, and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions, or products referred to in the content.

Article

A Two-Stage Deep Learning Framework for Uncertainty-Aware Forecasting and Conditional Process Optimization Toward Sustainable Smart Manufacturing

Junhee Lee ^{1,†} , Sajid Ali ^{1,†}  and Hail Jung ^{2,*} 

¹ Data Science Group, INTERX, Ulsan 44542, Republic of Korea

² Department of Business Administration, Seoul National University of Science and Technology, Seoul 01811, Republic of Korea

* Correspondence: hail95@seoultech.ac.kr; Tel.: +82-10-2994-7527

† These authors contributed equally to this work.

Abstract

The rise of Industry 4.0 has made robust forecasting and process control central to achieving sustainable smart manufacturing. While deep learning improves prediction accuracy, most models lack uncertainty quantification and process control—both critical for minimizing waste and optimizing energy use. Generative models are promising but underused in real-world manufacturing due to data and integration challenges. To address this, we propose a unified two-stage deep learning framework for real-time simulation and adaptive process control in dynamic manufacturing environments. The first stage leverages DeepAR and Monte Carlo Dropout to produce both accurate point predictions and calibrated uncertainty intervals. The second stage introduces a novel LSTM-based Conditional Variational Autoencoder (LSTM-CVAE) that reconstructs temporally coherent multivariate input sequences conditioned on target specifications. Then a two-stage filtering mechanism ensures the plausibility of generated inputs and their predictive alignment with operational goals. Experiments on two real-time industrial datasets (sugar and bioprocessing) and a public benchmark demonstrate average improvements of up to 35.7%, 21.0%, 24.6% in Balanced MAE, and 21.3%, 9.8%, 25.2% in Comprehensive Uncertainty Score metrics. The framework remains robust under batch-wise shuffling, supporting real-world deployment. Overall, it enables interpretable, forecast-aware, target-driven process control that improves resource efficiency and product cost—advancing sustainable manufacturing.

Keywords: Industry 4.0; artificial intelligence; uncertainty-aware forecasting; generative modeling; process control; sustainable manufacturing

1. Introduction

The emergence of Industry 4.0 has driven the convergence of cyber-physical systems, the Internet of Things (IoT), cloud computing, and big data analytics in modern manufacturing environments, generating large volumes of multivariate time-series data from machines, processes, and operating conditions [1–3]. These digital transformations enable advanced data-driven decision-making, which is critical not only for improving operational efficiency and product quality, but also for promoting sustainability through reduced energy consumption, waste minimization, and predictive maintenance. Leveraging recent advances in machine learning (ML) and deep learning (DL), Industrial AI has emerged as a powerful paradigm to extract actionable insights from complex manufacturing data [4,5]. However, realizing the full potential of these technologies remains challenging due to the high dimensionality, heterogeneity, and noise inherent in real-world industrial datasets [6]. Addressing these challenges is essential to support sustainable and intelligent manufacturing operations.

1.1. Background

Industry 4.0 builds upon the digital automation of Industry 3.0 by introducing interconnected, intelligent machines capable of decentralized decision-making and adaptive control without direct human input [7]. This transformation is largely driven by the integration of ML and DL algorithms, which enable manufacturing systems to perform complex tasks and adapt to dynamic conditions based on data-driven insights rather than predefined rules [8]. These algorithms are typically categorized into supervised and unsupervised learning, depending on the nature of data labeling and learning objectives [9].

In smart manufacturing, ML and DL enable four levels of analytics: descriptive, diagnostic, predictive, and prescriptive [10]. Descriptive analytics provides a summary of historical operations—e.g., Iftikhar et al. [11] visualized sensor data to identify downtime patterns. Diagnostic analytics uncovers root causes of issues, as in the CNN-LSTM-GRU model used by Han et al. [12] for detecting faults in rotating equipment. Predictive analytics forecasts future events such as Remaining Useful Life (RUL), supporting proactive maintenance strategies as demonstrated by Kang et al. [13]. Finally, prescriptive analytics recommends optimal actions to enhance outcomes, such as the reinforcement learning framework developed by Lepenioti et al. [14] for sustainable process control in steel manufacturing. Together, these analytics form the backbone of intelligent decision-making in operations, contributing not only to improved efficiency and quality but also to enhanced sustainability by reducing waste, energy consumption, and unplanned downtime.

Beyond traditional manufacturing, ML and DL have found broad applications in enabling sustainable practices across diverse industries. These include real-time decision-making in embedded AI systems [15], energy-aware adaptive factory systems [16], health and performance monitoring for human-centric manufacturing [17], optimization of energy systems for environmental sustainability [18], autonomous transportation for emission reduction [19], and AI-powered precision agriculture for maximizing yield with minimal environmental impact [20]. These examples demonstrate the critical role of AI technologies in advancing both operational excellence and sustainability across domains.

1.2. Issue

Despite the transformative potential of ML and DL in industrial settings, several barriers continue to limit their practical deployment in smart manufacturing environments. One major challenge is the organizational knowledge gap in AI: many manufacturing firms lack a clear understanding of how to collect, preprocess, and leverage high-value operational data, which undermines model reliability, scalability, and trustworthiness [21]. This gap poses a significant risk to achieving the goals of sustainable and resilient operations.

Traditional ML models often struggle to manage the dynamic and time-dependent nature of industrial processes [22,23]. A key concern is concept drift, where underlying data distributions evolve over time, leading to severe degradation in model performance once deployed. This disconnect between training and real-world conditions hinders effective decision-making and can result in wasteful or even unsafe operational outcomes [24]. Two primary obstacles restrict the effectiveness of ML/DL in production environments:

- **Insufficient data volume and quality:** Industrial data is often sparse, noisy, and inconsistent due to sensor malfunctions, heterogeneous sources, or environmental variability. Preprocessing is time-intensive, and accurate labeling is resource-intensive and error-prone [25,26]. These issues reduce model performance and impede sustainable insights generation.
- **Limited model adaptability and transferability:** ML/DL models trained for specific machines, production lines, or factories often fail to generalize across different settings. Overfitted models lack robustness, while overly generic models miss critical domain-specific dynamics. Furthermore, many existing forecasting models provide only point estimates, lacking mechanisms to quantify uncertainty—an essential capability in sustainability-critical or safety-sensitive applications [27].

Addressing these issues is critical to building intelligent, adaptive, and trustworthy decision-support systems that can contribute to sustainable manufacturing outcomes. To that end, robust frameworks are needed that:

- Adapt to concept drift while preserving model performance over time,
- Quantify predictive uncertainty to improve confidence in decisions and support risk-aware operations,
- Generalize effectively across diverse processes, equipment, and industrial contexts.

1.3. Our Idea

This study proposes a two-stage deep learning framework designed to enhance decision-making reliability and adaptability in dynamic manufacturing systems, with an emphasis on sustainable and resilient operations. The architecture integrates both predictive and generative components, structured as follows:

- **Uncertainty-aware forecasting module:** Provides calibrated point predictions along with probabilistic confidence intervals, enabling risk-aware decision support under dynamic conditions.
- **Conditional generative module (LSTM-CVAE):** Reconstructs realistic multivariate input trajectories conditioned on desired outcomes or target distributions, allowing controllable scenario generation.

Together, these modules empower manufacturing systems to capture temporal dynamics while simulating feasible operational pathways that align with target specifications or forecast ranges. By supporting adaptive and informed control strategies, the framework enables more efficient process optimization, reduces resource waste, and enhances operational robustness. These capabilities are particularly valuable for advancing sustainable manufacturing practices, where reliability, flexibility, and resource efficiency are essential to long-term performance.

1.4. Contributions

- **Multivariate Sequence Reconstruction with Forecast Consistency:** We propose an end-to-end deep learning framework that unifies uncertainty-aware probabilistic forecasting with multivariate sequence reconstruction. By recursively predicting outcomes from reconstructed inputs, the model ensures both point-wise accuracy and distributional reliability. Temporal dependencies are preserved, enabling the system to realistically capture process volatility. This conditional generation supports sustainable operations by reducing resource consumption, minimizing waste, and optimizing energy use.
- **Two-Stage Filtering for Sustainable Scenario Control:** We design a novel two-stage filtering mechanism that improves the realism and control of generated operational sequences. Sequences with high reconstruction errors are filtered out based on the learned MAE distribution, and the best-aligned sequence is selected based on target accuracy. This enhances scenario flexibility and decision controllability, supporting adaptive and resource-efficient production planning under uncertainty.
- **Temporal Robustness Under Disrupted Conditions:** Our framework demonstrates robustness to batch-order variations by preserving local temporal patterns rather than relying on fixed sequence order. This enables stable performance even under dynamically changing production schedules—an essential property for resilient, real-time decision-making in sustainable manufacturing environments.
- **Generalizability Across Sustainable Manufacturing Domains:** We validate the framework on diverse real-world datasets, including sugar and biomaterial manufacturing processes, as well as open-source benchmarks. The results confirm strong generalization performance, supporting its applicability across domains to enhance operational efficiency and contribute to sustainability goals.

2. Literature Review

The rapid digitization of manufacturing under Industry 4.0 has accelerated the use of machine learning (ML) and deep learning (DL) for process monitoring, quality control, and system optimization. However, as industrial systems become more dynamic and data-driven, traditional rule-based or static models struggle with adaptability, real-time performance, and scalability across heterogeneous environments. In response, recent research has explored ML/DL solutions that offer greater flexibility, predictive power, and resilience in manufacturing operations.

In the context of sustainable manufacturing, these advances hold promise not only for improving product quality and operational efficiency, but also for reducing energy consumption, minimizing waste, and enabling data-informed decision-making under uncertainty. This review highlights developments in three areas that form the foundation of our proposed approach: (1) ML/DL applications in smart manufacturing; (2) uncertainty-aware forecasting for quality management and operational risk mitigation; and (3) data-driven process optimization frameworks for adaptive and sustainable production systems.

2.1. ML and DL Applications in Manufacturing

Recent literature demonstrates the growing impact of machine learning (ML) and deep learning (DL) across a wide range of manufacturing applications, including anomaly detection, predictive maintenance, production planning, and quality control. These approaches are increasingly integrated with optimization techniques to improve decision-making at both machine and operator levels. For example, [28,29] offer comprehensive overviews of ML-based optimization strategies aimed at improving throughput and reducing downtime, while [30] highlights ongoing challenges related to model transferability, scalability, and interoperability across heterogeneous manufacturing systems.

In fault detection, deep learning models—particularly convolutional neural networks (CNNs) and recurrent architectures like LSTM—have achieved impressive accuracy rates of up to 97–100% in controlled environments [31]. Autoencoder-based anomaly detection has also been widely explored for real-time monitoring in Industry 4.0 settings [32]. Additionally, DL models have shown promise in predictive quality modeling, as discussed in [33], though deployment remains challenged by data inconsistencies, model drift, and integration complexity.

While these studies demonstrate the potential of ML/DL to enhance operational efficiency and responsiveness, their contributions to **sustainable manufacturing**—through waste reduction, energy optimization, and adaptive process control—remain less emphasized. Moreover, current models often lack the flexibility to adapt to new quality targets or operational constraints in real time, limiting their effectiveness in dynamic production environments.

Our work builds on this foundation by addressing the need for flexible, forecast-aware, and condition-driven control strategies that align predictive accuracy with sustainable operations goals.

2.2. Uncertainty Forecasting in Manufacturing Quality Management

In quality-critical manufacturing, probabilistic forecasting is essential for robust, risk-aware decision-making. While early methods focused on deterministic point estimates, recent research highlights the importance of uncertainty quantification to capture the variability inherent in industrial processes. Common approaches include Bayesian inference, Gaussian Process Regression (GPR), and Monte Carlo Dropout (MC Dropout), each balancing interpretability, scalability, and robustness differently.

For example, [34] used Bayesian SVR with Brownian modeling to forecast tool wear distributions for predictive maintenance, while [35] applied GPR to improve geometric accuracy in additive manufacturing via risk-aware tuning. [36] introduced an anomaly-based method for defect detection under limited labels. However, [37] reported that many uncertainty quantification methods degrade under distribution shift—a common issue in real-world production.

To overcome this, [38] introduced RoughLSTM, a robust deep learning architecture that incorporates temporal roughness constraints to improve anomaly detection in CNC vibration datasets. While promising, such Bayesian and uncertainty-aware forecasting models are still underutilized in manufacturing due to their complexity and the lack of real-time deployment frameworks.

Importantly, uncertainty-aware forecasting not only improves reliability and safety but also directly supports sustainability goals. By accounting for predictive confidence, manufacturers can optimize resource allocation, reduce overprocessing, and minimize energy and material waste through more targeted interventions.

Our proposed framework builds upon these insights by combining uncertainty-calibrated forecasting with conditional generative modeling, enabling real-time simulation and control tailored to specific operational targets or risk profiles. This integration enhances both predictive fidelity and decision adaptability—critical for sustainable manufacturing operations.

2.3. ML and DL Optimization in Manufacturing

Data-driven optimization techniques have significantly transformed manufacturing by enabling adaptive, real-time control without the need for explicit physical modeling. Machine learning (ML) and deep learning (DL) approaches, particularly reinforcement learning (RL), have been increasingly applied to improve process quality, reduce costs, and enhance system responsiveness.

Recent studies have applied reinforcement learning (RL) to optimize manufacturing processes, such as injection molding [39], additive manufacturing [40], forging [41], and textiles, achieving real-time control, quality improvement, and cost reduction. Approaches using DDPG, PPO, and SAC have shown significant speedups over heuristics [42] and support zero-defect and sustainable production goals [43].

While RL is effective for dynamic control, generative models offer a complementary approach by learning structured representations for controllable input generation. CVAEs and CGANs have been used for anomaly detection, data augmentation, and imputing missing sensor data [44–48]. However, their use for forecast-informed generation and simulation-based optimization remains underexplored in industrial applications.

These generative approaches hold strong potential for supporting sustainable manufacturing, particularly in cases requiring scenario simulation, adaptive control, and decision-making under uncertainty. By generating plausible input conditions aligned with target outcomes, they can help reduce material waste, improve process yield, and support energy-efficient production strategies.

Building on these foundations, our work explores the integration of uncertainty-aware forecasting with generative modeling to create a flexible, simulation-ready framework for target-driven process optimization—contributing directly to smart and sustainable manufacturing operations.

3. Methods

We propose an end-to-end framework for uncertainty-aware forecasting and target-conditioned input optimization in multivariate time-series manufacturing data. By jointly modeling prediction and generation, the framework reconstructs realistic input sequences that preserve temporal dependencies and align with probabilistic forecasts. This forecast-aware reconstruction (conditional generation) supports accurate, robust, and controllable process optimization under real-world variability which contributes to product cost and resource-efficient sustainable manufacturing.

Our pipeline consists of the following core components:

1. Prediction Model Specification (Section 3.1)

We implement probabilistic forecasting using:

- Monte Carlo Dropout (MCD) for DL models [49],
- Likelihood-based LSTM models (Simplified DeepAR) [50].

2. Forecast Evaluation and Predictor Selection (Section 3.2)

Model performance is evaluated using:

- MAE score[51] combining both the point accuracy and volatility sensitivity.
- Unified Uncertainty Score combining coverage (PICP) and sharpness (MPIW)[52,53].

These metrics are used for dual-model selection: one model is chosen for point forecasting, and another for uncertainty estimation.

3. Modified LSTM-Conditional VAE (Section 3.3)

We implement a Conditional Variational Autoencoder (CVAE) with LSTM layers [54,55]:

- The encoder augments each condition vector (target value) with recent statistical features from prior input sequences and learns latent distributions via KL annealing with target-aware penalties, thereby enhancing its ability to capture key temporal patterns.
- The decoder reconstructs multivariate sequences using sampled latent vectors and augmented condition vectors.
- To improve generalization and robustness, Gaussian noise is post-hoc injected into both the encoder and decoder.

4. Three-Stage Validation (Section 3.4)

To evaluate whether the reconstructed sequences preserve temporally meaningful and predictive patterns :

- Reconstruction Evaluation: Assess the temporal reconstruction quality by measuring MAE differences in point values, variability, and volatility.
- Downstream Forecast Evaluation: Use the best-trained point-wise and uncertainty-aware predictors to evaluate whether the reconstructed inputs $\mathbf{x}_{\text{recon}}$ preserve forecast-relevant dynamics by computing $\hat{y} = f^*(\mathbf{x}_{\text{recon}})$.
- Robustness Check: Test the model's ability to capture both local patterns and global dynamics by evaluating performance under batch-wise temporal shuffling—independent of strict sequential batch-to-batch order.

5. Multivariate Sequence Generation (Section 3.5)

Conditioned on a defined or forecasted target, the trained decoder generates multiple candidate sequences, which are then filtered through a two-stage selection process:

- Discard sequences with reconstruction error exceeding the MAE threshold observed during training.
- Select the sequence whose point-wise predicted target is closest to the conditioning value.

This process ensures that the selected candidate sequence maintains empirical validity while remaining consistent with the forecast target.

3.1. Prediction Model Specification

To implement probabilistic forecasting, two types of Deep Learning(DL) models are used.

1. Monte Carlo Dropout (MCD): Dropout [56] is applied at both training and inference to enable stochastic sampling. For each input, N forward passes generate multiple output samples. Uncertainty is computed as:

$$\hat{y} = \mu \pm n \cdot \sigma \quad \text{where } n \in \{1.65, 1.96, 2.58\} \quad (1)$$

Here, μ and σ denote the sample mean and standard deviation, and n is the confidence factor (e.g., $n = 1.96$ for 95%) [57]. MCD is implemented across various backbones such as LSTM and CNN-LSTM [58], enabling uncertainty-aware forecasting via stochastic inference.

2. Likelihood-Based Model (Simplified DeepAR): This model simplifies the original DeepAR by removing the autoregressive decoder and directly predicting the Gaussian distribution parameters (μ, σ) [59] using a single LSTM layer. Unlike the original recursive approach, all target distribution

parameters are inferred at once from the input sequence, eliminating the need for iterative decoding. The standard deviation σ is constrained to be positive via the softplus activation [60]:

$$\text{softplus}(x) = \log(1 + e^x) \quad (2)$$

The model is trained using the Gaussian negative log-likelihood (NLL) loss [61]:

$$\mathcal{L}_{\text{Gaussian NLL}} = \frac{1}{2} \log(2\pi) + \log(\sigma) + \frac{1}{2} \left(\frac{y - \mu}{\sigma} \right)^2 \quad (3)$$

The predicted mean μ serves as the point forecast, and the uncertainty interval is computed similarly to MCD (Eq. 1).

3.2. Forecast Measurements and Predictor Selection

3.2.1. Balanced Point Forecast Metric

To evaluate predictive performance in a more comprehensive manner, we employ a Balanced MAE Score, which jointly assesses absolute point-wise accuracy and the ability to follow local temporal volatility. The Balanced MAE Score consists of two components:

Absolute MAE — captures the average absolute deviation between predicted and true values (original MAE score).

Differential MAE — measures the accuracy in predicting the changes (first differences) in the time series.

Let y_i be the true value and \hat{y}_i be the predicted value. The Balanced MAE Score is defined as:

$$\text{Balanced-MAE} = \frac{1}{2} \left(\frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| + \frac{1}{n-1} \sum_{i=2}^n |\Delta y_i - \Delta \hat{y}_i| \right) \quad (4)$$

where $\Delta y_i = y_i - y_{i-1}$ and $\Delta \hat{y}_i = \hat{y}_i - \hat{y}_{i-1}$.

Lower values of the Balanced MAE indicate that the model not only produces accurate point predictions but also captures the underlying dynamic behavior.

3.2.2. Uncertainty Forecast Metric

To assess the distributional quality of predictive uncertainty, we evaluate the reliability of the prediction intervals using two metrics: Prediction Interval Coverage Probability (PICP) and Mean Prediction Interval Width (MPIW) which are the widely used in recent studies[62].

PICP measures the proportion of true values that lie within the predicted interval $[\hat{y}_i^L, \hat{y}_i^U]$, indicating coverage quality:

$$\text{PICP} = \frac{1}{n} \sum_{i=1}^n \mathbb{I} \left[\hat{y}_i^L \leq y_i \leq \hat{y}_i^U \right] \quad (5)$$

MPIW captures the average width of these intervals, reflecting how sharp or concentrated the predictions are:

$$\text{MPIW} = \frac{1}{n} \sum_{i=1}^n \left(\hat{y}_i^U - \hat{y}_i^L \right) \quad (6)$$

To jointly evaluate both aspects, we define a unified uncertainty score that balances high coverage and narrow interval width:

$$\text{Uncertainty Score} = \lambda \cdot (1 - \text{PICP}) + (1 - \lambda) \cdot \frac{\text{MPIW}}{y_{\max} - y_{\min}} \quad (7)$$

where $\lambda \in [0, 1]$ controls the trade-off between coverage and sharpness. A lower score indicates more reliable and sharper uncertainty estimates. In our work, we set $\lambda = 0.5$ to equally balance the prediction uncertainty measurements.

3.2.3. Dual Predictor Selection Strategy

For point and uncertainty forecasting, the model with the lowest Balanced-MAE and Unified Uncertainty Score on the test dataset is individually selected. This selection strategy ensures not only accurate point predictions but also robust and reliable uncertainty distribution quantification across diverse inference objectives.

3.3. Modified LSTM-Conditional VAE Architecture

We propose a modified LSTM-CVAE to effectively reconstruct (conditionally generate) multivariate time-series data conditioned on augmented target-related context.

3.3.1. Augmented Condition Vector

To enrich the condition vector (target value) $\mathbf{c}_t \in \mathbb{R}^D$, we extract feature-wise statistics from recent batch-wise sequences $\hat{\mathbf{x}}_{t-N:t-1} \in \mathbb{R}^{N' \times T \times F}$:

$$\mu_t = \frac{1}{N'} \sum_{i=t-N'}^{t-1} \hat{\mathbf{x}}_i, \quad \sigma_t = \sqrt{\frac{1}{N'} \sum_{i=t-N'}^{t-1} (\hat{\mathbf{x}}_i - \mu_t)^2} \quad (8)$$

If $N' = 0$, null vectors are used:

$$\mu_t = \mathbf{0}, \quad \sigma_t = \mathbf{0} \quad (9)$$

We then compute scalar indicators:

$$\bar{\mu}_t = \frac{1}{F} \sum_{j=1}^F [\mu_t]_j, \quad \bar{\sigma}_t = \frac{1}{F} \sum_{j=1}^F [\sigma_t]_j \quad (10)$$

The final augmented condition becomes:

$$\tilde{\mathbf{c}}_t = \text{concat}(\mathbf{c}_t, \bar{\mu}_t, \bar{\sigma}_t) \quad (11)$$

This allows the model to adapt to statistical temporal patterns (trend and volatility) with partially available sequences prior to each condition vector.

3.3.2. Conditional LSTM Encoder

To encode the input sequence, each time step \mathbf{x}_t is concatenated with the augmented condition vector $\tilde{\mathbf{c}}_t$:

$$\mathbf{x}'_{1:T} = [\mathbf{x}_1; \tilde{\mathbf{c}}_1], \dots, [\mathbf{x}_T; \tilde{\mathbf{c}}_T] \quad (12)$$

This results in a context-aware input sequence of shape $\mathbb{R}^{T \times (F+d_c)}$, where $d_c = \dim(\tilde{\mathbf{c}}_t) = D + 2$, as it includes the original condition vector and two scalar indicators. Gaussian noise [63] is post hoc added to $\tilde{\mathbf{c}}_t$ to improve robustness and avoid overfitting:

$$\tilde{\mathbf{c}}_t \leftarrow \tilde{\mathbf{c}}_t + \epsilon, \quad \epsilon \sim \mathcal{N}(0, \sigma_{\text{noise}}^2 \mathbf{I}) \quad (13)$$

The resulting sequence is passed through an LSTM layer to obtain a final hidden state \mathbf{h}_{enc} summarizing the temporal and conditional information:

$$\mathbf{h}_{\text{enc}} = \text{LSTM}(\mathbf{x}'_{1:T}) \quad (14)$$

The posterior parameters of the latent variable are then computed from \mathbf{h}_{enc} :

$$\boldsymbol{\mu} = \mathbf{W}_\mu \mathbf{h}_{\text{enc}} + \mathbf{b}_\mu, \quad \log \sigma^2 = \mathbf{W}_{\log \text{var}} \mathbf{h}_{\text{enc}} + \mathbf{b}_{\log \text{var}} \quad (15)$$

3.3.3. Latent Sampling via Reparameterization

To allow gradient flow during training, the reparameterization trick[64] samples latent variables \mathbf{z} from the approximate posterior:

$$\mathbf{z} = \boldsymbol{\mu} + \sigma \odot \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(0, \mathbf{I}) \quad (16)$$

where $\sigma = \exp(0.5 \cdot \log \sigma^2)$ ensures positive variance, and the element-wise product \odot introduces stochasticity in a differentiable way. This allows the encoder to produce a latent distribution rather than a point estimate, improving generalization [65].

3.3.4. Conditional LSTM Decoder

The latent vector \mathbf{z} is first projected into a high-dimensional hidden state using a non-linear transformation:

$$\mathbf{h}_z = \phi(\mathbf{W}_z \mathbf{z} + \mathbf{b}_z) \quad (17)$$

This transformed vector is repeated T times along the temporal axis to match the sequence length. The augmented condition vector $\tilde{\mathbf{c}}_t$ is also repeated and concatenated at each timestep to form the decoder input:

$$\mathbf{h}_t^{(0)} = [\mathbf{h}_{z,t}; \tilde{\mathbf{c}}_t], \quad \forall t = 1, \dots, T \quad (18)$$

This combined input is passed through an LSTM layer to model the conditional temporal dynamics:

$$\mathbf{h}_{1:T}^{\text{dec}} = \text{LSTM}(\mathbf{h}_{1:T}^{(0)}) \quad (19)$$

Finally, the decoder outputs are projected to reconstruct the original input sequence:

$$\hat{\mathbf{x}}_t = \mathbf{W}_{\text{out}} \cdot \mathbf{h}_t^{\text{dec}} + \mathbf{b}_{\text{out}}, \quad \forall t = 1, \dots, T \quad (20)$$

Gaussian noise is post hoc added to the reconstructed sequence to improve robustness and avoid overfitting:

$$\hat{\mathbf{x}}_{1:T} \leftarrow \hat{\mathbf{x}}_{1:T} + \boldsymbol{\epsilon}, \quad \boldsymbol{\epsilon} \sim \mathcal{N}(0, \sigma_{\text{noise}}^2 \mathbf{I}) \quad (21)$$

3.3.5. Loss Function with Target-Aware Penalty

During training, the encoder approximates the posterior $q(\mathbf{z} | [\mathbf{x}_t, \tilde{\mathbf{c}}_t])$ to sample the latent vector \mathbf{z} , and the decoder reconstructs \mathbf{x} via the conditional likelihood $p(\hat{\mathbf{x}}_t | [\mathbf{z}, \tilde{\mathbf{c}}_t])$.

The overall training objective consists of two components based on ELBO[66]: the reconstruction loss and the Kullback–Leibler (KL) divergence[67]. The reconstruction loss is defined as the mean absolute error (MAE) between the input and the reconstructed sequences:

$$\mathcal{L}_{\text{recon}} = \frac{1}{T} \sum_{t=1}^T \|\mathbf{x}_t - \hat{\mathbf{x}}_t\|_1 \quad (22)$$

The KL divergence regularizes the latent space by aligning the approximate posterior with a condition-aware prior. Specifically, it minimizes the divergence between $q(\mathbf{z} | [\mathbf{x}_t, \tilde{\mathbf{c}}_t])$ and $p(\mathbf{z} | \tilde{\mathbf{c}}_t)$:

$$\mathcal{L}_{\text{KL}} = D_{\text{KL}}(q(\mathbf{z} | \mathbf{x}_t, \tilde{\mathbf{c}}_t) \parallel p(\mathbf{z} | \tilde{\mathbf{c}}_t)) \quad (23)$$

To stabilize training, we apply KL annealing [68] with a weighting factor $\lambda_{\text{KL}} \in [0, 1]$, which gradually increases over epochs to avoid premature regularization of the latent space:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{recon}} + \lambda_{\text{KL}} \cdot \mathcal{L}_{\text{KL}} \quad (24)$$

To improve the quality of reconstructions for high-value targets (e.g., peak-relative or rare outlier patterns), we introduce a target-aware penalty that adaptively increases the training loss for such samples. If the condition value c_t exceeds a predefined threshold τ , a penalty weight $p \geq 1$ is applied to the total loss:

$$\mathcal{L}_{\text{weighted}} = (1 + \mathbb{I}(c_t \geq \tau) \cdot p) \cdot \mathcal{L}_{\text{total}} \quad (25)$$

where $\mathbb{I}(\cdot)$ is an indicator function that outputs 1 when the condition is satisfied and 0 otherwise. This allows the model to prioritize reconstruction quality for critical cases without sacrificing overall stability.

3.4. Three-Stage Validation

3.4.1. Reconstruction Evaluation

To match the original data format, the 3D input and reconstruction tensors (B, T, D) are flattened to 2D (B, F) , where $F = T \cdot D$, before evaluation.

The reconstruction quality using the MAE difference is assessed with three standards:

- **Point Values:**

$$\text{MAE} = \frac{1}{N \cdot F} \sum_{i=1}^N \sum_{f=1}^F |x_{i,f} - \hat{x}_{i,f}| \quad (26)$$

- **Standard Deviation (Variability):**

$$\text{Std}(|x - \hat{x}|) = \sqrt{\frac{1}{N \cdot F} \sum_{i=1}^N \sum_{f=1}^F \left(|x_{i,f} - \hat{x}_{i,f}| - \text{MAE} \right)^2} \quad (27)$$

- **First Differences (Volatility):**

$$\text{MAE}_{\Delta} = \frac{1}{N \cdot (F-1)} \sum_{i=1}^N \sum_{f=2}^F \left| (x_{i,f} - x_{i,f-1}) - (\hat{x}_{i,f} - \hat{x}_{i,f-1}) \right| \quad (28)$$

These metrics evaluate not only the formal closeness of reconstructed values, but also the temporal dynamics.

3.4.2. Downstream Forecast Evaluation

In addition, the reconstructed sequences are further evaluated by predicting their corresponding target values. While low reconstruction error indicates overall closeness to the original data, further evaluation is done to ensure that temporal patterns essential for point and uncertainty prediction are also well preserved. Accordingly, each reconstructed sequence $\mathbf{x}_{\text{recon}}$ is passed through the best predictors f^* to estimate the target and its associated uncertainty, as described in Section 3.2:

$$\hat{y} = f^*(\mathbf{x}_{\text{recon}}) \quad (29)$$

This downstream evaluation ensures that the reconstruction retains forecast-relevant dynamics, rather than merely optimizing reconstruction error.

3.4.3. Robustness Evaluation of Forecast-Aware Reconstruction (Conditional Generation) under Temporal Disruption

To evaluate the robustness of the proposed architecture under temporal shifts, a batch-wise shuffling experiment is applied. For each dataset, the batch axis B is randomly permuted while preserving the temporal structure within each input–target sequence pair $(\mathbf{x}^{(b)}, \mathbf{y}^{(b)})$. This disrupts inter-batch dependencies while maintaining intra-sequence dynamics within each pair. Given input–target pairs $(\mathbf{x}^{(1)}, \mathbf{y}^{(1)}), (\mathbf{x}^{(2)}, \mathbf{y}^{(2)}), \dots, (\mathbf{x}^{(B)}, \mathbf{y}^{(B)})$ where each $\mathbf{x}^{(b)} \in \mathbb{R}^{T \times D}$ and $\mathbf{y}^{(b)}$ is the associated target, a random permutation π is applied to the batch indices such that:

$$(\mathbf{x}, \mathbf{y})_{\text{shuffled}} = (\mathbf{x}^{(\pi(1))}, \mathbf{y}^{(\pi(1))}), \dots, (\mathbf{x}^{(\pi(B))}, \mathbf{y}^{(\pi(B))}) \quad (30)$$

where π is a random permutation of $1, 2, \dots, B$, uniformly sampled from all possible orders.

Evaluation metrics from Section 3.4.1 are recomputed on shuffled data using the best-performing fixed model identified in Section 3.4.2, without retraining. This setting assess the model's robustness under disrupted batch-level temporal order, highlighting its reliance on localized intra-sequence patterns. Each experiment is repeated R times per dataset, and results are reported as mean \pm standard deviation.

3.5. Multivariate Sequence Generation

Building on the trained LSTM-CVAE, we enable conditional generation of diverse multivariate time-series sequences that align with both target predictions and temporal dynamics captured during training.

3.5.1. Conditional Input Generation

Given a specific condition vector \mathbf{c}_t , the augmented condition vector $\tilde{\mathbf{c}}_t$ is configured using recent N batch-wise sequences, as described in Section 3.3.1. A latent vector \mathbf{z} is first randomly sampled, and the trained decoder concretizes this latent space to generate a candidate sequence $\hat{\mathbf{x}}$ from the conditional distribution $P(\mathbf{x} | [\mathbf{z}, \tilde{\mathbf{c}}_t])$. This process is repeated S times to enable diverse sampling under a fixed condition, and post hoc Gaussian noise is added to each generated sequence, as described in Section 3.3.4.

$$\mathbf{z}^{(j)} \sim \mathcal{N}(0, \mathbf{I}), \quad j = 1, \dots, S \quad (31)$$

$$\hat{\mathbf{x}}_{1:T}^{(j)} = \text{Decoder}(\mathbf{z}^{(j)}, \tilde{\mathbf{c}}), \quad j = 1, \dots, S_{\text{samples}} \quad (32)$$

$$\hat{\mathbf{x}}_{1:T}^{(j)} \leftarrow \hat{\mathbf{x}}_{1:T}^{(j)} + \boldsymbol{\varepsilon}, \quad \boldsymbol{\varepsilon} \sim \mathcal{N}(0, \sigma_{\text{noise}}^2 \mathbf{I}) \quad (33)$$

3.5.2. Two-Stage Filtering

The reconstruction loss of each decoder-generated candidate sequence is computed using the trained LSTM-CVAE, measured by the Mean Absolute Error (MAE) between the generated input and its corresponding reconstruction:

$$\text{MAE} = \frac{1}{N \cdot T \cdot F} \sum_{i=1}^N \sum_{t=1}^T \sum_{f=1}^F |x_{i,t,f} - \hat{x}_{i,t,f}| \quad (34)$$

where N is the number of sequences, T is the sequence length, and F is the number of features. $x_{i,t,f}$ and $\hat{x}_{i,t,f}$ represent the generated and reconstructed values, respectively, for sample i , time step t , and feature f .

To filter out unrealistic candidates, the reconstruction MAE of each generated sequence is normalized with respect to the distribution of reconstruction errors observed on the training set. Any

candidate whose MAE exceeds the maximum reconstruction error from the training distribution is discarded:

$$\hat{\mathbf{x}}^{(j)}; \text{ is discarded if; } \text{MAE}(\hat{\mathbf{x}}^{(j)}) > \max(\text{MAE}_{\text{train}}) \quad (35)$$

In the second stage, the remaining candidates are passed to the best-performing point predictor (Section 3.2.3). The predicted value for each sequence is compared with the target value c_{target} , and the closest-matching sequence is selected:

$$\hat{\mathbf{x}}_{1:T}^{(j^*)} = \arg \min_j |f(\hat{\mathbf{x}}_{1:T}^{(j)}) - c_{\text{target}}| \quad (36)$$

The final output $\hat{\mathbf{x}}_{1:T}^{(j^*)}$ is returned as the most realistic and target-consistent input sequence for process optimization.

4. Experiment Results

We evaluate the proposed architecture on three multivariate time-series datasets: two from real-world manufacturing sites and one from a public benchmark. The datasets span different manufacturing domains, ensuring result diversity and generalizability.

4.1. Data Collection and Preprocessing

We used two real-world datasets from sugar and bio manufacturing sites in South Korea, and a public Kaggle dataset¹ with noisy sensors. For the Kaggle data, the last 100,000 samples are used, targeting Sensor_02 (lag-1 autocorrelation $\rho = 0.9637$ [69]). Samples with target ≤ -5 are excluded, leaving 99,573 points. First-order differences ($x_t - x_{t-1}$) are added to capture volatility. Table 1 summarizes dataset domains, sizes, and targets.

Table 1. Summary of datasets used in this study, including their domain, size, dimensions, and target variables.

Dataset	Domain	Raw Size	Shape (Rows \times Cols)	Target Variable
A (Private)	Sugar	71,820	71,820 \times 29	Brix concentration
B (Private)	Bio	9,201	9,201 \times 15	pH concentration
C (Public)	Sensor	99,573	99,573 \times 25	Raw Sensor_02

Final dataset sizes and splits are shown in Table 2. Deep Learning model inputs follow the format $\mathbf{X} \in \mathbb{R}^{B \times T \times D}$, where B , T , and D denote batch size, sequence length, and feature dimension, respectively. Data is chronologically split into train, validation, and test sets. Feature-wise Min-Max scaling [70] is applied using training set statistics, with scaling ranges of $[-1, 1]$ for Datasets A and C, and $[0, 1]$ for Dataset B.

Table 2. Input and target data shapes across three datasets (A, B, and C), including train, validation, and test splits.

Dataset	Split	Input Shape	Target Shape	Shape Format
A	Train	59,980 \times 20 \times 28	59,980 \times 1	(B, T, D)
	Val	1,241 \times 20 \times 28	1,241 \times 1	(B, T, D)
	Test	10,000 \times 20 \times 28	10,000 \times 1	(B, T, D)
B	Train	6,415 \times 85 \times 14	6,415 \times 1	(B, T, D)
	Val	531 \times 85 \times 14	531 \times 1	(B, T, D)
	Test	2,000 \times 85 \times 14	2,000 \times 1	(B, T, D)
C	Train	83,940 \times 60 \times 24	83,940 \times 1	(B, T, D)
	Val	5,454 \times 60 \times 24	5,454 \times 1	(B, T, D)
	Test	10,000 \times 60 \times 24	10,000 \times 1	(B, T, D)

¹ <https://www.kaggle.com/competitions/tabular-playground-series-apr-2022/data>

4.2. Prediction Model Configuration and Results

Model settings are selected via validation. Architectures use ReLU [71] or TanH [72] activations with 0.1–0.3 dropout. LSTM/DeepAR employ 32–64 hidden units; CNN-LSTM models apply 1–2 convolutional layers (16–32 filters, kernel size 2–3, causal padding)[73] followed by dense layers. All models use Adam optimizer ($\text{lr} = 10^{-3}$)[74], with dataset-specific tuning of epochs, batch size, and early stopping [75].

As referenced in Section 3.1, $N = 100$ stochastic forward passes are used in MCD. The confidence factor n is set to 2.58 (99%) for Datasets A and B, and 1.65 (90%) for Dataset C. Table 3 summarizes the best-performing predictors for point forecasting and uncertainty estimation across datasets.

Table 3. Best-performing models for point prediction and uncertainty estimation across three datasets. It reports the top model for each task along with its respective performance score.

Dataset	Measure Type	Best Model	Result
A	Balanced-MAE	DeepAR	0.6673
	Unified Uncertainty Score		0.1776
B	Balanced-MAE	MCD CNN-LSTM	0.1742
	Unified Uncertainty Score	DeepAR	0.2549
C	Balanced-MAE	MCD CNN-LSTM	0.3178
	Unified Uncertainty Score	DeepAR	0.3306

4.3. Reconstruction Model Configuration and Results

Model configurations and parameters are selected based on validation datasets. Table 4 summarizes the parameters based on our proposed LSTM-CVAE. Model-specific variations include architecture, batch size, KL weight, target-aware threshold, penalty weight and Gaussian noise level. To avoid overfitting to rare peaks or extreme fluctuations, target-aware thresholds are set using the 95th percentile bounds of the training data. The penalty weight is fixed at 1.0 to ensure stable training while maintaining constraint effectiveness.

The Adam optimizer (learning rate 10^{-3}) and early stopping are consistently applied, with batch sizes adjusted per dataset. KL divergence is linearly annealed over the first 50 epochs, increasing from 0 to a dataset-specific final weight to stabilize latent learning and prevent posterior collapse.

Table 4. Reconstruction Model configuration details stage across three datasets.

Parameter	Dataset A	Dataset B	Dataset C
Model architecture	128–20–128	64–16–64	32–8–32
Best epoch (early stopping)	23/500	81/200	39/500
Batch size	64	32	64
Optimizer	Adam (learning rate 10^{-3})		
KL Weight	0.440	1.00	0.760
Validation MAE Loss	0.0227	0.0376	0.0030
Recent Batch Size (N)	10	10	30
Target-Aware threshold (τ)	Upper 95th Percentile	Upper 95th Percentile	Lower 95th Percentile
Penalty weight (p)	1.0	1.0	1.0
Encoder noise (Gaussian std σ)	0.5	0.1	0.001
Posthoc noise (Gaussian std σ)	0.015	0.2	0.01

For evaluation, we compared the proposed model with four representative generative baselines: standard Conditional VAE and CGAN [76], as well as their Wasserstein variants—Wasserstein-CVAE and Wasserstein-CGAN [77,78]. Additionally, a simple statistical baseline (Mean Reversion) is included by applying the mean of each variable from the test set. All generative models use the same LSTM architecture, training epochs, and batch size to ensure fair comparison.

Table 5 reports the forecast-aware reconstruction (conditional generation) performance of our method on Datasets A, B, and C compared to baseline approaches. The reconstruction metrics and

point/uncertainty forecast errors are evaluated as referenced in Section 3.4.1 and Section 3.4.2. In addition to forecasting accuracy, these results validate that the reconstructed inputs remain both plausible and target/uncertainty-consistent under injected noise. Notably, the forecasting performance based on reconstructed inputs closely aligns with the actual test results in Table 3, highlighting their reliability for uncertainty-aware prediction.

Although reconstruction errors are not always minimal, the reconstructed inputs consistently preserve temporal dynamics critical for downstream forecasting—even under encoder and decoder noise. This robustness highlights their ability to capture key volatility in test data, which is essential for realistic, target-driven condition generation. These results emphasize that in forecast-aware reconstruction, utility for downstream tasks is as important as reconstruction fidelity.

Figure 1 illustrates that reconstructed sequences not only produce accurate point predictions closely aligned with the true targets across Datasets A, B, and C, but also provide well-calibrated prediction intervals (PI) that capture the underlying distributional uncertainty of potential target values.

In contrast, Figure 2 presents the best-performing baseline predictions (excluding Mean Reversion) for each dataset. While these models generally capture overall trends, they tend to regress toward the mean and fail to represent local volatility—limiting their ability to reflect sharp predictive fluctuations. This highlights the advantage of our LSTM-CVAE in leveraging forecast-aware reconstruction (i.e., conditional sequence generation).

Taken together, these results highlight the robustness of our architecture in synthesizing realistic, context-aware sequences for downstream decision-making. The model serves as a reliable backbone for conditional process generation, producing tailored input sequences aligned with diverse probabilistic quality targets. By maintaining predictive consistency in reconstructed sequences, it enhances the reliability of condition-based simulation and optimization—enabling precise control over process trajectories in quality-sensitive manufacturing.

Table 5. Comparison of Forecast-Aware Reconstruction Performance Across Models on Three Datasets. Metrics include MAE and standard deviation of reconstruction error, MAE of first-order differences, and point/uncertainty prediction error. Best results are highlighted in bold.

Model	MAE($ x - \hat{x} $)	Std ($ x - \hat{x} $)	MAE($ \Delta x - \Delta \hat{x} $)	Point / Uncertainty Error
Dataset A				
Proposed	5.1387	11.4274	1.8455	0.7516/0.1805
Mean Reversion	4.8363	11.4224	1.1245	1.1228/0.2456
LSTM-CVAE	5.0352	11.9609	1.2786	1.01691/0.2582
LSTM-WCVAE	5.2271	12.0363	2.2277	1.1855/0.2801
LSTM-CGAN	12.6540	19.7733	4.1229	1.6908/0.4080
LSTM-WCGAN	6.2006	14.4517	5.5122	1.3956/0.3238
Dataset B				
Proposed	10.7326	15.7482	9.8223	0.0191/0.2604
Mean Reversion	8.1444	11.4462	0.3268	0.0304/0.3042
LSTM-CVAE	9.6854	12.8046	0.8718	0.0299/0.2970
LSTM-WCVAE	10.0363	12.9578	1.8012	0.0302/0.2915
LSTM-CGAN	28.0970	63.6527	11.5516	0.0391/0.3864
LSTM-WCGAN	11.2080	15.0513	5.9422	0.0308/0.3127
Dataset C				
Proposed	13.0457	27.4023	10.2044	0.4585/0.2948
Mean Reversion	9.3925	18.1818	3.6464	0.6926/0.3450
LSTM-CVAE	7.3437	10.2652	7.5356	0.5039/0.3060
LSTM-WCVAE	24.2246	55.4900	7.6082	0.5616/0.3831
LSTM-CGAN	27.6172	61.3529	7.0803	0.7971/0.5492
LSTM-WCGAN	10.8584	23.1551	12.3627	0.8194/0.4304

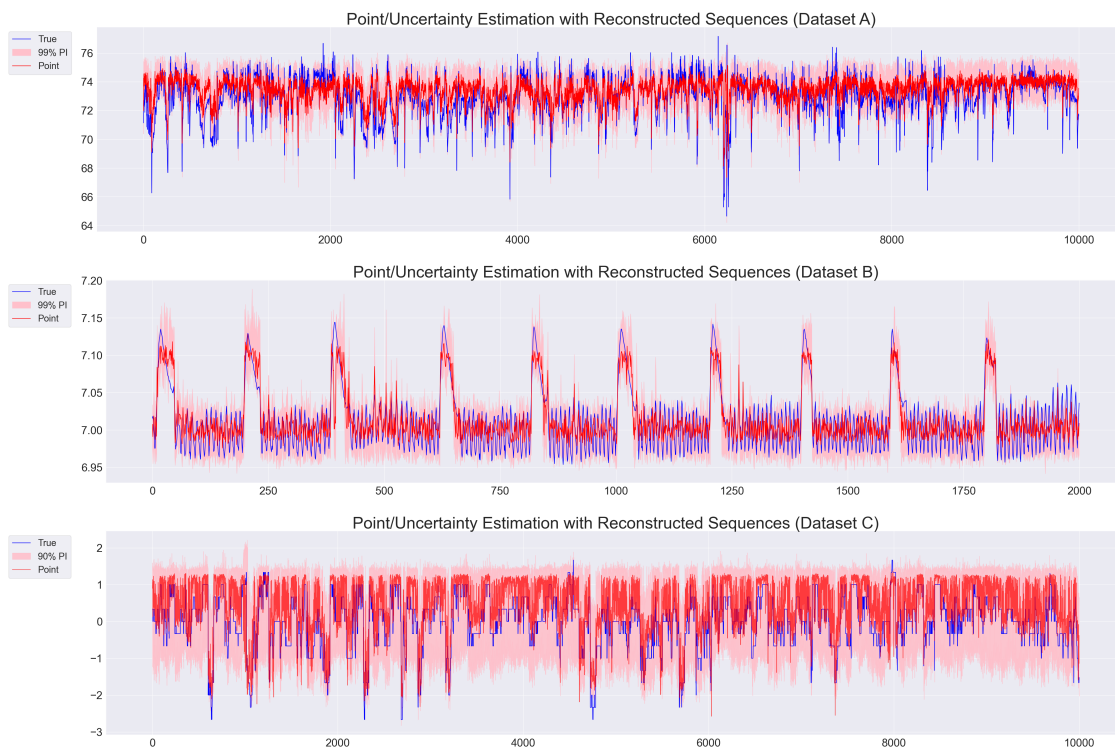


Figure 1. Forecast-Aware Reconstruction Results based on our proposed LSTM-CVAE across Datasets A (top), B (middle), and C (bottom). True targets (blue), predictive means (red), and prediction intervals (shaded) are shown.

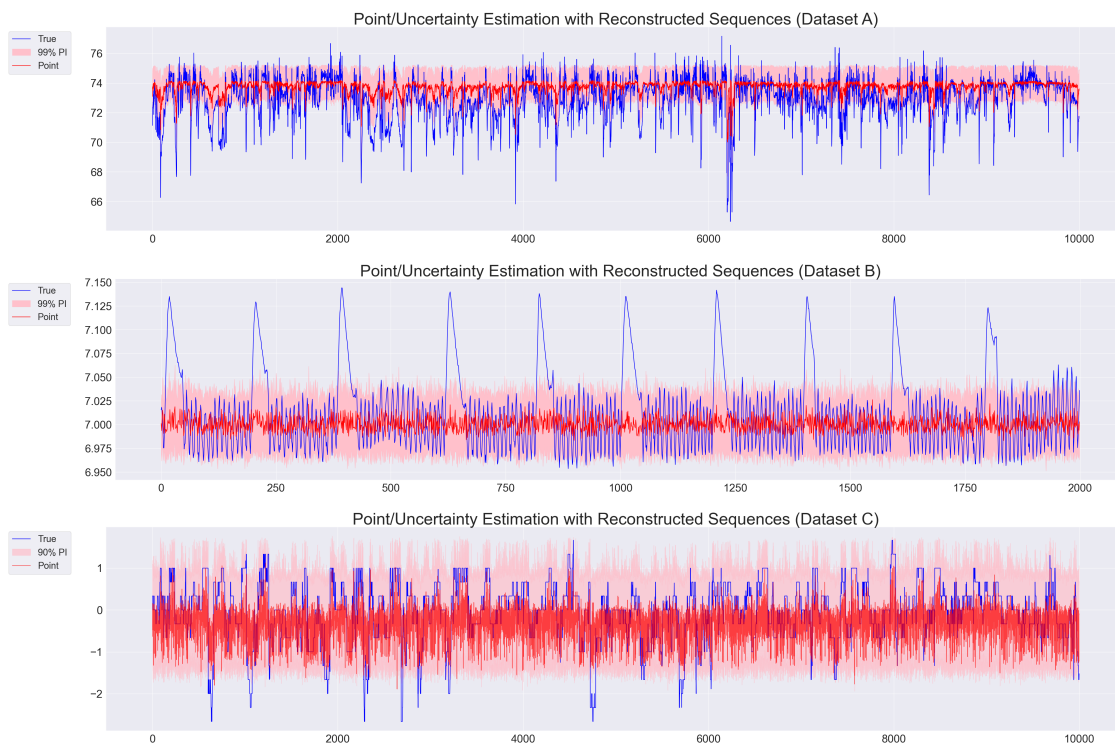


Figure 2. Forecast-Aware Reconstruction Results based on the best-performing baselines(excluding Mean Reversion) across Datasets A (top), B (middle), and C (bottom). True targets (blue), predictive means (red), and prediction intervals (shaded) are shown.

4.4. Evaluation Under Batch-Wise Temporal Shuffling

As referenced in Section 3.4.3, R is set to 5, and the results—based on the identical data shapes shown in Table 2—are presented in Table 6. Despite the disruption caused by data reordering—which makes these results not directly comparable to those in Table 5—point-wise and uncertainty estimation

remain stable across all datasets, although reconstruction MAE and Δx loss slightly increase under shuffling.

Notably, the uncertainty errors remain comparable to or even better than those in Table 5, indicating that our LSTM-CVAE can reliably infer target-quality distributions even without inter-batch temporal structure. Using the fixed best-performing predictor from Section 4.2, the model effectively captures robust intra-sequence dynamics, enabling flexible process generation without strictly relying on batch-level temporal constraints.

Table 6. Evaluation of Batch-wise Temporal Shuffling on Reconstruction and Forecasting Performance Across Different Datasets.

Dataset	MAE($ x - \hat{x} $)	Std ($ x - \hat{x} $)	MAE($ \Delta x - \Delta \hat{x} $)	Point / Uncertainty Error
A	6.6684 ± 0.14	15.8906 ± 0.14	9.3369 ± 0.017	$0.4164 \pm 0.02 / 0.2197 \pm 0.009$
B	10.6756 ± 0.38	15.5924 ± 0.44	14.9615 ± 0.14	$0.01632 \pm 1.94 \times 10^{-5} / 0.3047 \pm 0.003$
C	18.25 ± 0.7	40.82 ± 1.78	13.54 ± 0.28	$0.4993 \pm 0.053 / 0.3054 \pm 0.013$

4.5. Multivariate Sequence Generation Evaluation

Building on the predictive robustness established earlier, the practical utility of our LSTM-CVAE is evaluated in generating multivariate sequences conditioned on both probabilistic forecasts and user-defined targets. This highlights the model's ability to produce realistic, target-aligned sequences for simulation and optimization in quality-sensitive manufacturing settings.

As referenced in Section 3.5.1, the recent N batches per dataset (Table 4) are selected from the test set (Table 2) and used to compute point-wise and uncertainty-based targets with fixed forecasting models (Table 3). These targets form the condition vector c_t for sequence generation in forecasting-driven scenario simulation.

Each dataset uses the augmented condition vector \tilde{c}_t to generate $S = 1000$ candidate sequences via the trained decoder. To expand coverage, minimum and maximum target values are additionally incorporated for simulating user-defined scenarios beyond the forecast range.

Following the two-stage selection process (Section 3.5.2), sequences below the MAE threshold are retained and the one best matching each target c_t is selected. As shown in Table 7, generated sequences remain coherent—even under relatively extreme targets (e.g., minimum or maximum values)—demonstrating reliable target alignment.

Table 7. Performance of Candidate Sequence Generation Across Datasets A, B, and C using various target conditions (c_t). Metrics include maximum training MAE, number of sequences passing filtering, target value, best sequence index, and corresponding prediction.

Dataset	Target Type	max(MAE _{train})	Candidate Seqs	c_t	Best Seq Index	Best Prediction
A	Point	0.4054	822 / 1,000	74.25	674	74.2492
	Lower		812 / 1,000	72.10	720	72.7877
	Upper		846 / 1,000	75.00	4	75.0135
	Min		867 / 1,000	60.08	499	65.32
	Max		877 / 1,000	77.37	868	75.83
B	Point	0.3037	899 / 1,000	7.0016	22	7.0015
	Lower		898 / 1,000	6.9700	281	6.9732
	Upper		899 / 1,000	7.1300	593	7.1076
	Min		898 / 1,000	6.9535	897	6.9726
	Max		898 / 1,000	7.1527	592	7.1260
C	Point	0.1271	981/1000	-1.11	886	-1.1123
	Lower		970/1000	-2.10	719	-2.0997
	Upper		987/1000	1.05	838	1.0468
	Min		953/1000	-4.90	57	-4.0312
	Max		989/1000	1.67	649	1.3597

To further evaluate robustness, all test batches are randomly shuffled, and sequences are re-generated from the last N batches per dataset. Despite disrupted temporal order, comparable

results (Table 8) confirm that our architecture captures localized intra-sequence patterns, consistently producing realistic, context-aware sequences across both forecast-based and user-defined scenarios.

Table 8. Performance of Augmented Condition-Aware Sequence Generation Across Multiple Target Types and Datasets (Batchwise-Shuffled).

Dataset	Target Type	max(MAE _{train})	Candidate Seqs	c_t	Best Seq Index	Best Prediction
A	Point	0.4054	781 / 1,000	74.25	351	74.2512
	Lower		784 / 1,000	72.10	559	72.0962
	Upper		796 / 1,000	75.00	140	74.9187
	Min		792 / 1,000	60.08	44	65.3882
	Max		753 / 1,000	77.37	671	75.7628
B	Point	0.3037	869 / 1,000	7.0016	655	7.0015
	Lower		898 / 1,000	6.9700	868	6.9746
	Upper		871 / 1,000	7.1300	189	7.1240
	Min		868 / 1,000	6.9535	769	6.9747
	Max		816 / 1,000	7.1527	176	7.1356
C	Point	0.1271	980/1000	-1.11	162	-1.1083
	Lower		967/1000	-2.10	541	-2.0978
	Upper		985/1000	1.05	142	1.0501
	Min		953/1000	-4.90	583	-4.1495
	Max		986/1000	1.67	647	1.3514

5. Discussion

This section provides insights into how the proposed framework achieves structural preservation, conditional generation, and robustness—capabilities that are essential for sustainable and adaptive manufacturing operations.

As shown in Table 5, the LSTM-CVAE does not always yield the lowest reconstruction error. However, it more effectively retains temporal dynamics such as trend direction and volatility, which are critical for downstream forecasting and real-time decision support. Figure 1 illustrates this behavior, showing how forecast-aligned reconstructions follow both the target trajectory and the spread of the predictive distribution. This contrasts with baseline models (Figure 2), which tend to regress toward the mean and overlook process variability—limiting their usefulness in complex industrial contexts.

These advantages stem from key architectural components, including the augmented condition vector, KL-regularization guided by target-awareness, and post-hoc noise injection. Together, these elements promote the generation of plausible, data-consistent inputs that support realistic process simulations and enable flexible control strategies.

Robustness under changing temporal structures is another critical consideration in manufacturing environments, where batch order and sensor timing may not be consistent. As validated by Table 6, the model maintains stable performance even when the temporal order of test batches is disrupted. This indicates that the system captures local intra-sequence dependencies rather than relying on rigid batch-to-batch sequencing, making it suitable for dynamic, real-world deployment.

Additionally, Tables 7 and 8 confirm the model's ability to generate input trajectories that align with both probabilistic forecasts and user-defined targets. This conditional generation supports target-driven scenario simulation and process reconfiguration—essential tools in smart manufacturing where process control must be both accurate and resource-efficient.

Overall, these findings validate the core design principle of this work: embedding predictive relevance within the reconstruction process enables more than just statistical accuracy—it enables interpretable, controllable, and realistic simulation of future process states. This capability lays a foundation for scenario-based optimization that adapts to changing conditions, supports energy conservation, reduces waste, and enhances system resilience.

By enabling simulation-driven decision-making with forecast consistency and robust sequence generation, the proposed framework serves as a viable tool for smart, sustainable manufacturing operations. It empowers practitioners to optimize for performance while accounting for variability

and uncertainty—key factors in achieving environmentally responsible and operationally resilient industrial systems.

6. Conclusion

As modern manufacturing systems grow increasingly complex and data-rich, the ability to predict, adapt, and optimize in real time has become essential for achieving both operational efficiency and sustainable outcomes [79]. Moreover, AI and machine learning enhance production efficiency and sustainability by enabling data-driven optimization of complex processes—reducing energy use, increasing throughput, and minimizing inefficiencies. These gains support cost-effective operations and resource efficiency, demanding technical robustness and alignment with business goals [80].

This study proposes a two-stage deep learning framework that integrates uncertainty-aware forecasting with conditional sequence generation, built upon an LSTM-CVAE architecture. The model supports both inference ($X \rightarrow Y$) and generative reasoning ($Y \rightarrow X$), enabling realistic, simulation-ready scenarios that align with probabilistic forecasts or user-defined targets. By combining predictive accuracy with controllable input reconstruction, the framework facilitates adaptive process optimization under dynamic operating conditions.

Experimental results across multiple real-world datasets demonstrate that the model maintains critical temporal structures such as trend direction and local volatility, regardless of reconstruction error levels. This capability supports more robust forecasting and decision-making, especially in environments where data is noisy or production sequences are irregular. A two-stage filtering mechanism further enhances the realism and reliability of generated scenarios by selecting sequences based on reconstruction plausibility and forecast alignment with target distributions.

Importantly, the proposed framework contributes to sustainable manufacturing by enabling energy-aware and resource-efficient process adjustments in response to variable quality or performance targets. It allows practitioners to simulate, evaluate, and control manufacturing scenarios in real time, thereby supporting intelligent scheduling, proactive maintenance, and reduced environmental impact.

However, several limitations remain. The system relies on carefully tuned hyperparameters [81], and its Gaussian-based uncertainty modeling may limit expressiveness in highly skewed or multimodal settings. The filtering mechanism also lacks formal guarantees under extreme distributional shifts.

Future work could explore alternative uncertainty modeling techniques to improve adaptability and expressiveness. Reinforcement learning [82] may further enhance the sequence selection process, while transformer-based architectures [83] offer promising avenues for handling streaming data and long-range dependencies.

In summary, this research introduces a robust and extensible framework that bridges forecasting with condition-aware sequence generation to support real-time simulation, optimization, and adaptive control. By improving predictive fidelity and supporting operational flexibility, the proposed system offers a viable pathway toward smart, resilient, and sustainable manufacturing operations.

Author Contributions: Conceptualization, J.L., S.A. and H.J. ; methodology, J.L. , S.A. and H.J. ; software, J.L.; validation, J.L and S.A ; formal analysis, J.L. and S.A. ; investigation, J.L. and S.A. ; resources, J.L. ; data curation, J.L.; writing—original draft , J.L. and S.A. ; writing—review and editing, J.L. , S.A. and H.J. ; visualization, J.L. ; supervision, H.J. ; project administration, H.J. ; funding acquisition, H.J. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by Seoul National University of Science and Technology.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data are not publicly available due to their containing information that could compromise the privacy of research participants.

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

References

1. Lu, Y. Industry 4.0: A survey on technologies, applications and open research issues. *Journal of industrial information integration* **2017** *6*, 1-10.
2. Nizam, H.; Zafar, S.; Lv, Z.; Wang, F.; Hu, X. Real-time deep anomaly detection framework for multivariate time-series data in industrial IoT. *IEEE Sensors Journal*, **2022**, *22.23*: 22836-22849.
3. Yao, Y.; Yang, M.; Wang, J.; Xie, M. Multivariate time-series prediction in industrial processes via a deep hybrid network under data uncertainty. *IEEE Transactions on Industrial Informatics*, **2022**, *19.2*: 1977-1987.
4. Yadav, D. Machine learning: Trends, perspective, and prospects. *Science*, **2020**, *349*: 255-260.
5. Gupta, C.; Farahat, A. Deep learning for industrial AI: Challenges, new methods and best practices. In: *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. **2020**. p. 3571-3572.
6. Xu, J.; Kovatsch, M.; Mattern, D.; Mazza, F.; Harasic, M.; Paschke, A.; Lucia, S. A review on AI for smart manufacturing: Deep learning challenges and solutions. *Applied Sciences*, **2022**, *12.16*: 8239.
7. Lee, J.; Davari, H.; Singh, J.; Pandhare, V. Industrial Artificial Intelligence for industry 4.0-based manufacturing systems. *Manufacturing letters*, **2018**, *18*: 20-23.
8. Bousdekis, A.; Lepenioti, K.; Apostolou, D.; Mentzas, G. A review of data-driven decision-making methods for industry 4.0 maintenance applications. *Electronics*, **2021**, *10.7*: 828.
9. Alloghani, M.; Al-Jumeily, D.; Mustafina, J.; Hussain, A.; Aljaaf, A. J. A systematic review on supervised and unsupervised machine learning algorithms for data science. *Supervised and unsupervised learning for data science*, **2020**, *3*-21.
10. Roy, D.; Srivastava, R.; Jat, M.; Karaca, M. S. A complete overview of analytics techniques: descriptive, predictive, and prescriptive. *Decision intelligence analytics and the implementation of strategic business management*, **2022**, *15*-30.
11. Iftikhar, N.; Baattrup-Andersen, T.; Nordbjerg, F. E.; Bobolea, E.; Radu, P. B. Data Analytics for Smart Manufacturing: A Case Study. In: *DATA*. **2019**. p. 392-399.
12. Han, K.; Wang, W.; Guo, J. Research on a Bearing Fault Diagnosis Method Based on a CNN-LSTM-GRU Model. *Machines*, **2024**, *12.12*: 927.
13. Kang, Z.; Catal, C.; Tekinerdogan, B. Remaining useful life (RUL) prediction of equipment in production lines using artificial neural networks. *Sensors*, **2021**, *21.3*: 932.
14. Lepenioti, K.; Pertselakis, M.; Bousdekis, A.; Louca, A.; Lampathaki, F.; Apostolou, D.; Anastasiou, S. Machine learning for predictive and prescriptive analytics of operational data in smart manufacturing. In: *Advanced Information Systems Engineering Workshops: CAiSE 2020 International Workshops, Grenoble, France, June 8–12, 2020, Proceedings 32*. Springer International Publishing, **2020**. p. 5-16.
15. Zhao, R.; Yan, R.; Chen, Z.; Mao, K.; Wang, P.; Gao, R. X. Deep learning and its applications to machine health monitoring. *Mechanical Systems and Signal Processing*, **2019**, *115*: 213-237.
16. Awodiji, T. O. Industrial Big Data Analytics and Cyber-Physical Systems for Future Maintenance & Service Innovation. In: *CS & IT Conference Proceedings*. CS & IT Conference Proceedings, **2021**.
17. Attal, F.; Mohammed, S.; Dedabrishvili, M.; Chamroukhi, F.; Oukhellou, L.; Amirat, Y. Physical human activity recognition using wearable sensors. *Sensors*, **2015**, *15.12*: 31314-31338.
18. Muzaffar, S.; Afshari, A. Short-term load forecasts using LSTM networks. *Energy Procedia*, **2019**, *158*: 2922-2927.
19. Grigorescu, S.; Trasnea, B.; Cocias, T.; Macesanu, G. A survey of deep learning techniques for autonomous driving. *Journal of field robotics*, **2020**, *37.3*: 362-386.
20. Kotsiopoulos, T.; Sarigiannidis, P.; Ioannidis, D.; Tzovaras, D. Machine learning and deep learning in smart manufacturing: The smart grid paradigm. *Computer Science Review*, **2021**, *40*: 100341.
21. Plathottam, S. J.; Rzonca, A.; Lakhnori, R.; Iloeje, C. O. A review of artificial intelligence applications in manufacturing operations. *Journal of Advanced Manufacturing and Processing*, **2023**, *5.3*: e10159.
22. Wuest, T.; Weimer, D.; Irgens, C.; Thoben, K. D. WUEST, Thorsten, et al. Machine learning in manufacturing: advantages, challenges, and applications. *Production & Manufacturing Research*, **2016**, *4.1*: 23-45.
23. Shi, Y.; Wei, P.; Feng, K.; Feng, D. C.; Beer, M. A survey on machine learning approaches for uncertainty quantification of engineering systems. *Machine Learning for Computational Science and Engineering*, **2025**, *1.1*: 1-39.
24. Yong, B. X.; Brintrup, A. Bayesian autoencoders with uncertainty quantification: Towards trustworthy anomaly detection. *Expert Systems with Applications*, **2022**, *209*: 118196.

25. Escobar, C. A.; McGovern, M. E.; Morales-Menendez, R. Quality 4.0: a review of big data challenges in manufacturing. *Journal of Intelligent Manufacturing*, **2021**, 32.8: 2319-2334.
26. Xie, J.; Sun, L.; Zhao, Y. F. On the data quality and imbalance in machine learning-based design and manufacturing—A systematic review. *Engineering*, **2025**, 45.2: 105-131.
27. Serradilla, O.; Zugasti, E.; Rodriguez, J.; Zurutuza, U. Deep learning models for predictive maintenance: a survey, comparison, challenges and prospects. *Applied Intelligence*, **2022**, 52.10: 10934-10964.
28. Chatterjee, S.; Chaudhuri, R.; Vrontis, D.; Papadopoulos, T. Examining the impact of deep learning technology capability on manufacturing firms: moderating roles of technology turbulence and top management support. *Annals of Operations Research*, **2024**, 339.1: 163-183.
29. Weichert, D.; Link, P.; Stoll, A.; Rüping, S.; Ihlenfeldt, S.; Wrobel, S. A review of machine learning for the optimization of production processes. *The International Journal of Advanced Manufacturing Technology*, **2019**, 104.5: 1889-1902.
30. Ramesh, K.; Indrajith, M. N.; Prasanna, Y. S.; Deshmukh, S. S.; Parimi, C.; Ray, T. Comparison and assessment of machine learning approaches in manufacturing applications. *Industrial Artificial Intelligence*, **2025**, 3.1: 2.
31. Kim, S.; Seo, H.; Lee, E. C. Advanced Anomaly Detection in Manufacturing Processes: Leveraging Feature Value Analysis for Normalizing Anomalous Data. *Electronics*, **2024**, 13.7: 1384.
32. Liu, W.; Yan, L.; Ma, N.; Wang, G.; Ma, X.; Liu, P.; Tang, R. Unsupervised deep anomaly detection for industrial multivariate time series data. *Applied Sciences*, **2024**, 14.2: 774.
33. Tercan, H.; Meisen, T. Machine learning and deep learning based predictive quality in manufacturing: a systematic review. *Journal of Intelligent Manufacturing*, **2022**, 33.7: 1879-1905.
34. Zhang, H.; Jiang, S.; Gao, D.; Sun, Y.; Bai, W. A Review of Physics-Based, Data-Driven, and Hybrid Models for Tool Wear Monitoring. *Machines*, **2024**, 12.12: 833.
35. Hermann, F.; Michalowski, A.; Brünnette, T.; Reimann, P.; Vogt, S.; Graf, T. Data-driven prediction and uncertainty quantification of process parameters for directed energy deposition. *Materials*, **2023**, 16.23: 7308.
36. Incorvaia, G.; Hond, D.; Asgari, H. Uncertainty quantification of machine learning model performance via anomaly-based dataset dissimilarity measures. *Electronics*, **2024**, 13.5: 939.
37. Kompa, B.; Snoek, J.; Beam, A. L. Empirical frequentist coverage of deep learning uncertainty quantification procedures. *Entropy*, **2021**, 23.12: 1608.
38. Çekik, R.; Turan, A. Deep Learning for Anomaly Detection in CNC Machine Vibration Data: A RoughLSTM-Based Approach. *Applied Sciences*, **2025**, 15.6: 3179.
39. Khoudi, A.; Masrouf, T.; El Hassani, I.; El Mazgualdi, C. A deep-reinforcement-learning-based digital twin for manufacturing process optimization. *Systems*, **2024**, 12.2: 38.
40. Dharmadhikari, S.; Menon, N.; Basak, A. A reinforcement learning approach for process parameter optimization in additive manufacturing. *Additive Manufacturing*, **2023**, 71: 103556.
41. Ma, Y.; Kessler, A.; Ahmed, B. S.; Krakhmalev, P.; Thore, A.; Toyser, A.; Lindbäck, H. Using deep reinforcement learning for zero defect smart forging. In: *SPS2022*. IOS Press, **2022**. p. 701-712.
42. Kim, J. Y.; Yu, J.; Kim, H.; Ryu, S. DRL-Based Injection Molding Process Parameter Optimization for Adaptive and Profitable Production. *arXiv preprint arXiv:2505.10988*, **2025**.
43. Malashin, I.; Martysyuk, D.; Tynchenko, V.; Gantimurov, A.; Semikolenov, A.; Nelyub, V.; Borodulin, A. Machine Learning-Based Process Optimization in Biopolymer Manufacturing: A Review. *Polymers*, **2024**, 16.23: 3368.
44. Chung, J.; Shen, B.; Kong, Z. J. Anomaly detection in additive manufacturing processes using supervised classification with imbalanced sensor data based on generative adversarial network. *Journal of Intelligent Manufacturing*, **2024**, 35.5: 2387-2406.
45. Harford, S.; Karim, F.; Darabi, H. Generating adversarial samples on multivariate time series using variational autoencoders. *IEEE/CAA Journal of Automatica Sinica*, **2021**, 8.9: 1523-1538.
46. Bashar, M. A.; Nayak, R. Time series anomaly detection with adjusted-lstm gan. *International Journal of Data Science and Analytics*, **2025**, 1-19.
47. Feng, C. A cGAN Ensemble-based Uncertainty-aware Surrogate Model for Offline Model-based Optimization in Industrial Control Problems. In: *2024 International Joint Conference on Neural Networks (IJCNN)*. IEEE, **2024**. p. 1-8.
48. Liu, Q.; Cai, P.; Abueidda, D.; Vyas, S.; Koric, S.; Gomez-Bombarelli, R.; Geubelle, P. Univariate conditional variational autoencoder for morphogenic pattern design in frontal polymerization-based manufacturing. *Computer Methods in Applied Mechanics and Engineering*, **2025**, 438: 117848.

49. Gal, Y.; Ghahramani, Z. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In: *international conference on machine learning*. PMLR, **2016**. p. 1050-1059.
50. Salinas, D.; Flunkert, V.; Gasthaus, J.; Januschowski, T. DeepAR: Probabilistic forecasting with autoregressive recurrent networks. *International journal of forecasting*, **2020**, 36.3: 1181-1191.
51. Hodson, T. O. Root mean square error (RMSE) or mean absolute error (MAE): When to use them or not. *Geoscientific Model Development Discussions*, **2022**, 2022: 1-10.
52. Gneiting, T.; Balabdaoui, F.; Raftery, A. E. Probabilistic forecasts, calibration and sharpness. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, **2007**, 69.2: 243-268.
53. Pearce, T.; Brintrup, A.; Zaki, M.; Neely, A. High-quality prediction intervals for deep learning: A distribution-free, ensembled approach. In: *International conference on machine learning*. PMLR, **2018**. p. 4075-4084.
54. Suh, S.; Chae, D. H.; Kang, H. G.; Choi, S. Echo-state conditional variational autoencoder for anomaly detection. In: *2016 International Joint Conference on Neural Networks (IJCNN)*. IEEE, **2016**. p. 1015-1022.
55. Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural computation*, **1997**, 9.8: 1735-1780.
56. Nitish, S. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, **2014**, 15.1: 1929-1958.
57. Bland, J. M.; Altman, D. G. Transformations, means, and confidence intervals. *BMJ: British Medical Journal*, **1996**, 312.7038: 1079.
58. Lu, W.; Li, J.; Li, Y.; Sun, A.; Wang, J. A CNN-LSTM-based model to forecast stock prices. *Complexity*, **2020**, 2020.1: 6622927.
59. Zhang, X. Gaussian distribution. In: *Encyclopedia of machine learning and data mining*. Springer, Boston, MA, **2016**. p. 1-5.
60. Zheng, H.; Yang, Z.; Liu, W.; Liang, J.; Li, Y. Improving deep neural networks using softplus units. In: *2015 International joint conference on neural networks (IJCNN)*. IEEE, **2015**. p. 1-4.
61. Yao, H.; Zhu, D. L.; Jiang, B.; Yu, P. Negative log likelihood ratio loss for deep neural network classification. In: *Proceedings of the Future Technologies Conference (FTC) 2019: Volume 1*. Springer International Publishing, **2020**. p. 276-282.
62. Dufva, J. Machine Learning-Based Uncertainty Quantification for Postmortem Interval Prediction from Metabolomics Data. **2025**.
63. Lee, D. U.; Luk, W.; Villasenor, J. D.; Cheung, P. Y. A Gaussian noise generator for hardware-based simulations. *IEEE Transactions on Computers*, **2004**, 53.12: 1523-1534.
64. Luchnikov, I. A.; Ryzhov, A.; Stas, P. J.; Filippov, S. N.; Ouerdane, H. Variational autoencoder reconstruction of complex many-body physics. *Entropy*, **2019**, 21.11: 1091.
65. Pinheiro Cinelli, L.; Araújo Marins, M.; Barros da Silva, E. A.; Lima Netto, S. Variational autoencoder. In: *Variational methods for machine learning with applications to deep networks*. Cham: Springer International Publishing, **2021**. p. 111-149.
66. Blei, D. M.; Kucukelbir, A.; McAuliffe, J. D. Variational inference: A review for statisticians. *Journal of the American statistical Association*, **2017**, 112.518: 859-877.
67. Joyce, J. M. Kullback-leibler divergence. In: *International encyclopedia of statistical science*. Springer, Berlin, Heidelberg, **2011**. p. 720-722.
68. Fu, H.; Li, C.; Liu, X.; Gao, J.; Celikyilmaz, A.; Carin, L. Cyclical annealing schedule: A simple approach to mitigating kl vanishing. *arXiv preprint arXiv:1903.10145*, **2019**.
69. Hansen, P. R.; Lunde, A. Estimating the persistence and the autocorrelation function of a time series that is measured with error. *Econometric Theory*, **2014**, 30.1: 60-93.
70. Cabello-Solorzano, K.; Ortigosa de Araujo, I.; Peña, M.; Correia, L.; J. Tallón-Ballesteros, A. The impact of data normalization on the accuracy of machine learning algorithms: a comparative analysis. In: *International conference on soft computing models in industrial and environmental applications*. Cham: Springer Nature Switzerland, **2023**. p. 344-353.
71. Banerjee, C.; Mukherjee, T.; Pasilio Jr, E. An empirical study on generalizations of the ReLU activation function. In: *Proceedings of the 2019 ACM Southeast Conference*. **2019**. p. 164-167.
72. Zamanlooy, B.; Mirhassani, M. Efficient VLSI implementation of neural networks with hyperbolic tangent activation function. *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, **2013**, 22.1: 39-48.
73. Naseri, H.; Mehrdad, V. Novel CNN with investigation on accuracy by modifying stride, padding, kernel size and filter numbers. *Multimedia Tools and Applications*, **2023**, 82.15: 23673-23691.
74. Zhang, Z. Improved adam optimizer for deep neural networks. In: *2018 IEEE/ACM 26th international symposium on quality of service (IWQoS)*. Ieee, **2018**. p. 1-2.

75. Ji, Z.; Li, J.; Telgarsky, M. Early-stopped neural networks are consistent. *Advances in Neural Information Processing Systems*, **2021**, 34: 1805-1817.
76. Smith, K. E.; Smith, A. O. Conditional GAN for timeseries generation. *arXiv preprint arXiv:2006.16477*, **2020**.
77. Panaretos, V. M.; Zemel, Y. Statistical aspects of Wasserstein distances. *Annual review of statistics and its application*, **2019**, 6.1: 405-431.
78. Tolstikhin, I.; Bousquet, O.; Gelly, S.; Schoelkopf, B. Wasserstein auto-encoders. *arXiv preprint arXiv:1711.01558*, **2017**.
79. Lee, S. K.; Ko, H. Generative Machine Learning in Adaptive Control of Dynamic Manufacturing Processes: A Review. *arXiv preprint arXiv:2505.00210*, **2025**.
80. Lee, J.; Jang, J.; Tang, Q.; Jung, H. LEE, Junhee, et al. Recipe Based Anomaly Detection with Adaptable Learning: Implications on Sustainable Smart Manufacturing. *Sensors*, **2025**, 25.5: 1457.
81. Tynchenko, V.; Kukartseva, O.; Tynchenko, Y.; Kukartsev, V.; Panfilova, T.; Kravtsov, K.; Malashin, I. Predicting Tilapia Productivity in Geothermal Ponds: A Genetic Algorithm Approach for Sustainable Aquaculture Practices. *Sustainability*, **2024**, 16.21: 9276.
82. Huang, W.; Cui, Y.; Li, H.; Wu, X. Effective probabilistic neural networks model for model-based reinforcement learning USV. *IEEE Transactions on Automation Science and Engineering*, **2025**.
83. Kang, H.; Kang, P. Transformer-based multivariate time series anomaly detection using inter-variable attention mechanism. *Knowledge-Based Systems*, **2024**, 290: 111507.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.