Statistical Theory and Related Fields

ISSN: 2475-4269 (Print) 2475-4277 (Online) Journal homepage: www.tandfonline.com/journals/tstf20

Hybrid Kolmogorov-Arnold and xLSTM network for enhanced RUL prediction of lithium batteries

Yunhao Hu, Qiuyi Ge, Ziqing Tian, Zuwei Zhang & Fode Zhang

To cite this article: Yunhao Hu, Qiuyi Ge, Ziqing Tian, Zuwei Zhang & Fode Zhang (03 Mar 2026): Hybrid Kolmogorov-Arnold and xLSTM network for enhanced RUL prediction of lithium batteries, Statistical Theory and Related Fields, DOI: 10.1080/24754269.2026.2635735

To link to this article: https://doi.org/10.1080/24754269.2026.2635735

Published online: 03 Mar 2026.

STATISTICAL THEORY AND RELATED FIELDS
https://doi.org/10.1080/24754269.2026.2635735

Hybrid Kolmogorov-Arnold and xLSTM network for enhanced RUL prediction of lithium batteries

Yunhao Hu, Qiuyi Ge, Ziqing Tian, Zuwei Zhang and Fode Zhang

Centerof Statistical Research, School of Statistics and Data Science, Southwestern University of Finance and Economics, Chengdu, People's Republic of China

ABSTRACT

Accurate prediction of the remaining useful life (RUL) of lithium batteries is essential for ensuring efficient equipment maintenance and energy management, particularly as these batteries serve as a core driver in the new energy technology revolution. While deep learning models such as Convolutional Neural Networks (CNNs), Long Short-Term Memory Networks (LSTMs), and their variants have demonstrated significant success in RUL prediction, they often face challenges related to inadequate modelling of long-term dependencies in complex degradation data. To overcome these limitations, this paper proposes a novel hybrid architecture that integrates the Kolmogorov-Arnold Network (KAN) with an Extended Long Short-Term Memory Network (xLSTM). The KAN component enhances high-dimensional function approximation and improves parameter efficiency by substituting traditional linear weights with *B*-spline-parameterized univariate functions. Meanwhile, the xLSTM introduces exponential gating mechanisms and covariance update rules to more effectively capture high-order long-term dependencies. Experimental results on the NASA lithium battery aging dataset demonstrate that the proposed KAN-xLSTM model significantly outperforms CNN, LSTM, and xLSTM models in prediction accuracy, particularly for battery groups with large capacity fluctuations.

ARTICLE HISTORY
Received 28 November 2025
Revised 12 January 2026
Accepted 12 February 2026

KEYWORDS
RUL prediction; degradation data; Kolmogorov-Arnold network; extended LSTM

1. Introduction

The prediction of RUL is a key task in reliability engineering, system safety, prognostics, and health management (Lai et al., 2025), which estimates the time left before a system or component fails, based on observed data and degradation patterns (Cai et al., 2024). Lithium-ion batteries have become the preferred energy storage solution for a wide range of applications, including electric vehicles, renewable energy systems, and consumer electronics (Peiseler et al., 2024).

Their high energy density, long cycle life, and environmental advantages make them a key enabler of the clean energy transition (R. Zhu, Hu et al., 2025). However, ensuring the reliability and safety of lithium batteries remains a significant challenge, particularly as they undergo degradation during usage (Gao et al., 2025; N. Li et al., 2025).

Accurate prediction of RUL is essential for effective battery health management, enabling timely maintenance, reducing unexpected failures, and extending battery lifespan (J. Liu

CONTACT Fode Zhang fredzh@swufe.edu.cn Center of Statistical Research, School of Statistics and Data Science, Southwestern University of Finance and Economics, Chengdu 611130, People's Republic of China

et al., 2024). RUL prediction involves estimating the time or number of cycles a battery can continue to operate before reaching a critical performance threshold. This task is inherently complex due to the nonlinear and stochastic nature of battery degradation, which is influenced by factors such as temperature, discharge rate, and usage patterns. Traditional model-based approaches, including probability models and empirical models, often require stronger assumptions and prior knowledge (S. Zhang et al., 2023). On the other hand, data-driven methods, particularly those based on machine learning and deep learning, have gained significant attention for their ability to capture intricate degradation patterns from historical battery data, which need extensive domain knowledge and can be computationally intensive (K. Liu, Shang et al., 2021).

Deep learning models such as CNNs and LSTMs have shown promising results in RUL prediction by effectively extracting features from time series data and modelling temporal dependencies (Ren et al., 2018). However, these models often face challenges in capturing long-term dependencies and may suffer from overfitting, especially when dealing with noisy or limited data. Additionally, computational inefficiency remains a concern in large-scale applications. See the following literature review for more details.

1.1. Literature review

Previous research on remaining useful life (RUL) prediction can be mainly categorized into model-based and data-driven approaches (Hess et al., 2005). Model-based methods, grounded in mathematical or physical principles, construct predictive models by exploring degradation mechanisms. For instance, adaptive Brownian motion frameworks (Bu et al., 2025) and nonlinear Wiener process models incorporating component interactions (Niu et al., 2025) have been developed to enhance accuracy and model complex degradation. Data-driven approaches, in contrast, learn degradation patterns directly from historical monitoring data without requiring explicit prior knowledge of the underlying mechanisms. This category encompasses a wide range of techniques, including filtering algorithms (S. Li et al., 2021; Y. Li et al., 2021), stochastic process models (Bu et al., 2025; K. Liu, Hu et al., 2019), and more robust frameworks like the Student-t process for heavy-tailed data (Xu et al., 2025). Methods for online parameter updating (H. Wang et al., 2019) and model selection without a true degradation model (F. Zhang et al., 2023) have also been proposed.

In recent years, deep learning (DL) methods have demonstrated remarkable progress in RUL prediction (Y. Zhu et al., 2026), often surpassing traditional machine learning (Perumal et al., 2024). DL approaches excel at capturing complex temporal patterns and nonlinear relationships in degradation data. For example, recent advancements include federated learning frameworks for collaborative modelling under privacy constraints (R. Zhu, Peng et al., 2025), cross-domain transfer learning models for unlabelled target data (Shu et al., 2024), and domain generalization methods to improve performance under unknown operational conditions (J. Wang et al., 2024).

The Long Short-Term Memory (LSTM) network (Hochreiter & Schmidhuber, 1997), with its gating mechanism, remains a cornerstone for modelling temporal dependencies in RUL prediction. Variants such as Bidirectional LSTM (F. K. Wang et al., 2022) and attention-enhanced LSTM models like TPA-LSTM (Cui et al., 2021) and LSTM-P-A (Xiang, Qin, Zhu et al., 2020) have been developed to improve feature weighting and prediction accuracy. The Multicellular LSTM (MCLSTM) further addresses information updating by employing data-level partitioning (Xiang, Qin, Luo et al., 2021).

Two particularly relevant architectural innovations are the Kolmogorov-Arnold Network (KAN) (Z. Liu et al., 2024) and the extended LSTM (xLSTM) (Beck et al., 2024). KAN replaces traditional linear weights with learnable univariate functions parameterized by *B*-splines, offering high parameter efficiency and strong nonlinear approximation capabilities, which have proven effective in time series modelling (Vaca-Rubio et al., 2024). Hybrids like TKAN combine KAN with LSTM gating mechanisms (Genet & Inzirillo, 2025). Meanwhile, xLSTM introduces exponential gating and a matrix memory with a covariance update rule, enabling parallel computation while preserving LSTM's advantages in capturing long-term dependencies. These innovations directly motivate the hybrid KAN-xLSTM architecture proposed in this work, which aims to leverage the high approximation capability of KAN and the parallelizable long-term modelling of xLSTM for superior RUL prediction.

1.2. Contributions of this work

To address the above challenges, this paper proposes a novel hybrid architecture, KAN-xLSTM, which integrates the Kolmogorov-Arnold Network (KAN) with the Extended Long Short-Term Memory network (xLSTM). This integration is grounded in two key theoretical insights. First, KAN leverages the Kolmogorov-Arnold representation theorem by using *B*-spline-parameterized learnable univariate functions in place of traditional linear weights. This grants superior flexibility and high-dimensional function approximation capability. Second, xLSTM introduces exponential gating mechanisms and covariance update rules within a matrix-memory structure, enabling parallel computation and enhanced modelling of high-order long-term dependencies in sequential data. The proposed KAN-xLSTM synergistically combines these strengths, providing a powerful framework tailored for the nonlinear, multi-timescale nature of battery degradation modelling.

The main contributions of this work are summarized as follows:

(1) The proposed KAN-xLSTM integrates the high-dimensional approximation capability of KAN with the parallelizable long-term dependency modelling of xLSTM. By replacing linear weights with *B*-spline-parameterized functions, it achieves higher parameter efficiency and computational flexibility, addressing the limitations of traditional neural networks in handling nonlinear and multi-scale degradation patterns.
(2) The architecture integrates Kolmogorov-Arnold representation theorem and covariance update mechanisms, unifying rapid neural scaling laws with spatiotemporal dynamic modelling capabilities. This provides a solid mathematical foundation and modelling framework for battery degradation processes characterized by strong nonlinearity and multiple time scales.
(3) A modular hybrid architecture is proposed, in which KAN layers replace the linear transformation layers in xLSTM, achieving a unification of parameter efficiency and functional flexibility. This design not only enhances the model's ability to express complex degradation patterns but also supports efficient training through parallelizable mechanisms, offering a new structural paradigm for high-dimensional time series modelling.
(4) The framework establishes a new analytical paradigm for battery health management, enabling real-time RUL prediction with enhanced interpretability. It offers actionable insights for maintenance scheduling and energy optimization, bridging the gap between theoretical advancements and industrial applications.

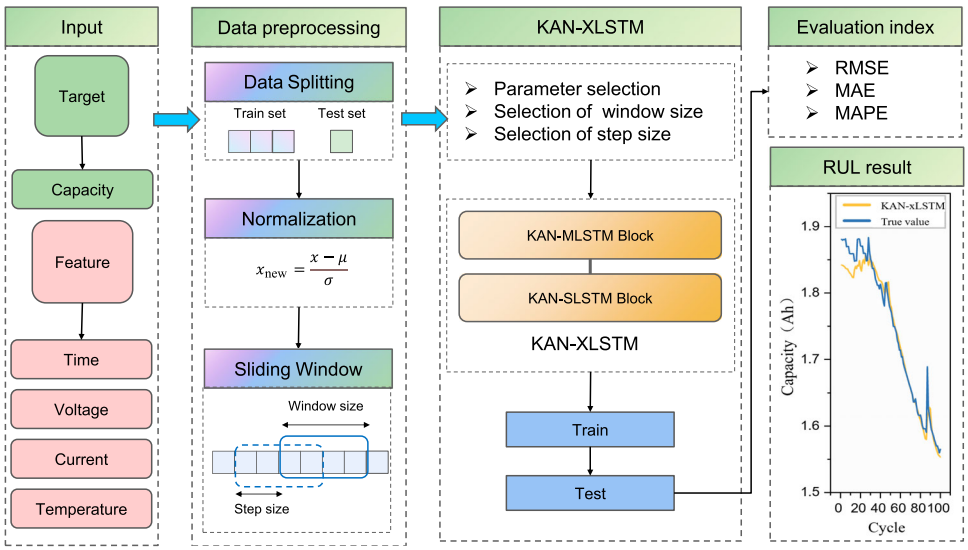

Figure 1. The framework of KAN-xLSTM model with applications to RUL prediction.

(5) Extensive validation on the NASA dataset demonstrates that KAN-xLSTM outperforms state-of-the-art models (CNN, LSTM) in RMSE, MAE, and MAPE metrics, particularly excelling in scenarios with large capacity fluctuations. The model's robustness is further confirmed through cross-validation and ablation studies.

The subsequent chapters are arranged as follows: Section 2 presents the theoretical framework and principles of KAN-xLSTM; Section 3 details the experimental design and comparative analysis; and the final section summarizes the research results and outlines future directions. The flowchart of this paper is shown below in Figure 1.

2. xLSTM with KAN memory units

2.1. Kolmogorov-Arnold networks

Multi-Layer Perceptrons (MLPs) (Hornik et al., 1989), an extension of the original perceptron proposed by Rosenblatt (1958), were inspired by the universal approximation theorem (Cybenko, 1989). This theorem states that a feedforward network (FFN) with a single hidden layer of finite neurons can approximate any continuous function on a compact subset of $\mathbb{R}^n$ arbitrarily well. In contrast, KANs are based on the Kolmogorov-Arnold representation theorem (Kolmogorov, 1961), which asserts that any multivariate continuous function f can be expressed as a composition of univariate functions through additive operations:

$$f(x_1, \ldots, x_n) = \sum_{q=1}^{2n+1} \Phi_q \left(\sum_{p=1}^{n} \phi_{q,p}(x_p) \right), \tag{1}$$

where, $\phi_{q,p}$ are univariate functions mapping $x_p \in [0,1]$ to $\mathbb{R}$, and $\Phi_q : \mathbb{R} \to \mathbb{R}$. Since these functions are univariate, they can be parameterized as B-spline curves, with learnable coefficients linked to local B-spline basis functions.

The similarity between MLPs and KANs lies in their layer structures. An MLP layer consists of a linear transformation followed by nonlinear operations, while a KAN layer is defined as

$$\Phi = \{\phi_{q,p}\}, \quad p = 1, 2, \ldots, n_{\text{in}}, \quad q = 1, 2, \ldots, n_{\text{out}}, \tag{2}$$

where $\phi_{q,p}$ are parametrized functions. In the Kolmogorov-Arnold theorem, the inner functions form a KAN layer with $n_{\text{in}} = n$ and $n_{\text{out}} = 2n + 1$, while the outer functions form another KAN layer with $n_{\text{in}} = 2n + 1$ and $n_{\text{out}} = 1$. Thus, the representation in Equation (1) is a composition of two KAN layers.

A deep Kolmogorov-Arnold representation extends this concept. A KAN is defined by its shape $[n_0, n_1, \ldots, n_L]$, where n_i denotes the number of nodes in the i-th layer (Z. Liu et al., 2024). The (l, i)-th neuron in layer l has an activation value $x_{l,i}$. Between layers l and $l+1$, there are activation functions $n_l \times n_{l+1}$ $\phi_{l,j,i}$, connecting the neuron (l, i) to $(l+1, j)$. The pre-activation is $x_{l,i}$, and the post-activation is $\tilde{x}_{l,j,i} = \phi_{l,j,i}(x_{l,i})$. The activation of neuron $(l+1, j)$ is

$$x_{l+1,j} = \sum_{i=1}^{n_l} \tilde{x}_{l,j,i} = \sum_{i=1}^{n_l} \phi_{l,j,i}(x_{l,i}), \quad j = 1, \ldots, n_{l+1}. \tag{3}$$

In matrix form, this becomes

$$x_{l+1} = \begin{pmatrix} \phi_{l,1,1}(\cdot) & \phi_{l,1,2}(\cdot) & \cdots & \phi_{l,1,n_l}(\cdot) \\ \phi_{l,2,1}(\cdot) & \phi_{l,2,2}(\cdot) & \cdots & \phi_{l,2,n_l}(\cdot) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_{l,n_{l+1},1}(\cdot) & \phi_{l,n_{l+1},2}(\cdot) & \cdots & \phi_{l,n_{l+1},n_l}(\cdot) \end{pmatrix} x_l, \tag{4}$$

where Φ_l is the function matrix for the l-th KAN layer. As Figure 2 shows, a general KAN model is a composition of L such layers, with the output given by

$$\text{KAN}(x) = (\Phi_{L-1} \circ \Phi_{L-2} \circ \cdots \circ \Phi_1 \circ \Phi_0)x. \tag{5}$$

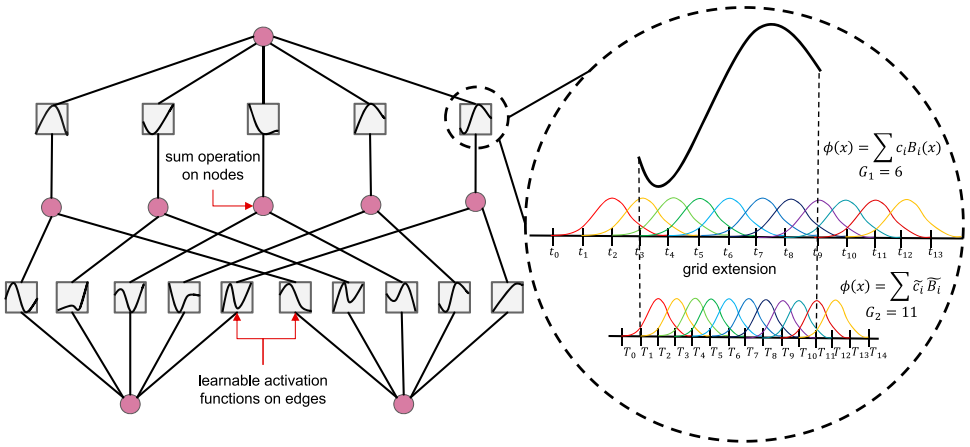

Figure 2. An illustration of KAN structure.

In contrast, our abstraction of KAN layers and their visualizations is cleaner and intuitive. The original Kolmogorov-Arnold representation in Equation (5) corresponds to a 2-Layer KAN with shape $[n, 2n + 1, 1]$. Notice that all the operations are differentiable, so the paper can train KANs with backpropagation.

2.2. xLSTM

The xLSTM model is built by stacking two basic components: sLSTM (Scaler Memory LSTM) and mLSTM (Matrix Memory LSTM), and embedding a residual structure. Here is an introduction to these components.

2.2.1. sLSTM

In the realm of recurrent neural networks, the Long Short-Term Memory (LSTM) architecture stands out for its ability to capture long-range dependencies. The weight vectors w_z, w_i, w_f, and w_o are crucial components, representing the input weight vectors between inputs x_t and the cell input, input gate, forget gate, and output gate, respectively. Similarly, the recurrent weights r_z, r_i, r_f, and r_o govern the connections between the hidden state h_{t-1} and these gates. The bias terms b_z, b_i, b_f, and b_o further fine-tune the network's performance. The activation functions ϕ, typically hyperbolic tangent (tanh), play vital roles in normalizing the cell input and hidden state. The gate activation functions are sigmoid functions, ensuring values between 0 and 1.

In more advanced LSTM formulations, multiple scalar memory cells $c_t \in \mathbb{R}^d$ are combined into a vector. This allows the use of recurrent weight matrices $R \in \mathbb{R}^{d\times d}$, which mix the outputs of memory cells, enhancing the network's ability to process complex sequences (Greff et al., 2016). Ablation studies have demonstrated that all components of the memory cell are essential for the LSTM's effectiveness.

To further enhance LSTMs, this paper introduces exponential gates alongside normalization and stabilization mechanisms. Specifically, input and forget gates can utilize exponential activation functions. For normalization, a normalizer state is introduced, which sums the product of the input gate and all future forget gates. This innovation empowers LSTMs with the ability to revise storage decisions dynamically, improving their adaptability and performance in handling sequential data.

The sLSTM improves on the traditional LSTM by introducing exponential gating, normalization, and stabilization mechanisms. The forward propagation process is described as follows:

$$c_t = f_t c_{t-1} + i_t z_t, \tag{1}$$

$$n_t = f_t n_{t-1} + i_t, \tag{2}$$

$$h_t = o_t \odot \tilde{h}_t, \quad \tilde{h}_t = c_t / n_t, \tag{3}$$

$$z_t = \varphi(\tilde{z}_t), \quad \tilde{z}_t = W_z^\top x_t + r_z h_{t-1} + b_z, \tag{4}$$

$$i_t = \exp(\tilde{i}_t), \quad \tilde{i}_t = W_i^\top x_t + r_i h_{t-1} + b_i, \tag{5}$$

$$f_t = \sigma(\tilde{f}_t) \text{ or } \exp(\tilde{f}_t), \quad \tilde{f}_t = W_f^\top x_t + r_f h_{t-1} + b_f, \tag{6}$$

$$o_t = \sigma(\tilde{o}_t), \quad \tilde{o}_t = W_o^\top x_t + r_o h_{t-1} + b_o. \tag{6}$$

The original LSTM gating techniques, i.e., input- and/or hidden-dependent gating plus bias terms, are transferred to the new architectures. Exponential activation functions can lead to large values that cause overflows. Therefore, for the numerical calculation of the stable exponential gating, the additional state m_t is introduced (Milakov & Gimelshein, 2018):

$$m_t = \max\left(\log(f_t) + m_{t-1}, \log(i_t)\right), \tag{7}$$

$$i'_t = \exp\left(\log(i_t) - m_t\right) = \exp(\tilde{i}_t - m_t), \tag{8}$$

$$f'_t = \exp\left(\log(f_t) + m_{t-1} - m_t\right). \tag{7}$$

The sLSTM, like the original LSTM, can have multiple memory cells. These cells enable memory mixing through recurrent connections R_z, R_i, R_f, R_o from the hidden state vector h to the input of the memory cell z and the gates i, f, o. A novel aspect of this memory mixing is the effect of exponential gating. The new sLSTM can have multiple heads, each with its own memory mixing, but without mixing across heads. The introduction of heads for sLSTM, along with exponential gating, creates a new method of memory mixing.

2.2.2. mLSTM

To enhance storage capacities of LSTMs, the paper increases the LSTM memory cell from a scalar $c \in \mathbb{R}$ to a matrix $C \in \mathbb{R}^{d \times d}$, which stores more contextual information that captures the complex relationships and dependencies in the input data. The covariance update rule (Dayan & Willshaw, 1991; Sejnowski, 1977) for storing a key-value pair is

$$C_t = C_{t-1} + v_t k_t^\top. \tag{8}$$

The paper assumes a layer-norm before projecting inputs to keys and values, so they have zero mean. The covariance update rule is optimal (Krotov & Hopfield, 2016) for maximal separability of retrieved binary vectors, which means a maximal signal-to-noise ratio. Higher separability is possible when limiting retrieval to pairwise interactions and accepting quadratic complexity like attention (Krotov & Hopfield, 2018; Ramsauer et al., 2021; Schmidhuber, 1992). The covariance update rule is similar to Fast Weight Programmers (Ba et al., 2016; Schlag et al., 2021), which later added a constant decay rate multiplied to C_{t-1} and a constant learning rate multiplied to $v_t k_t^\top$ (Sun et al., 2023). Following this idea, the covariance update rule is integrated into the LSTM framework. Here, the forget gate acts like a decay rate, the input gate like a learning rate, and the output gate scales the retrieved vector.

For this matrix memory, the normalizer state is the weighted sum of key vectors. Each key vector is weighted by the input gate and all future forget gates. The normalizer state records the strength of the gates. Since the dot product between the query and the normalizer state can be close to zero, the absolute value of this dot product is used, setting a lower bound (typically 1.0) as done previously (Beck et al., 2024). The mLSTM forward propagation process is given as follows:

$$C_t = f_t C_{t-1} + i_t v_t k_t^\top, \quad n_t = f_t n_{t-1} + i_t k_t,$$

$$h_t = o_t \odot \tilde{h}_t, \quad \tilde{h}_t = \frac{C_t q_t}{\max\left\{|n_t^\top q_t|, 1\right\}},$$

$$
\begin{aligned}
q_t &= W_q x_t + b_q, \quad k_t = \frac{1}{\sqrt{d}} W_k x_t + b_k, \\
v_t &= W_v x_t + b_v, \quad i_t = \exp(\tilde{i}_t), \quad \tilde{i}_t = W_i^\top x_t + b_i, \\
f_t &= \sigma(\tilde{f}_t) \text{ or } \exp(\tilde{f}_t), \quad \tilde{f}_t = W_f^\top x_t + b_f, \\
o_t &= \sigma(\tilde{o}_t), \quad \tilde{o}_t = W_o x_t + b_o.
\end{aligned} \tag{9}
$$

In mLSTM, multiple memory cells can be incorporated, akin to the original LSTM. For mLSTM, multiple heads and multiple cells are equivalent concepts since there is no memory mixing. The stabilization techniques used for sLSTM (referenced in Equation (7)) are also applied to stabilize the exponential gates of mLSTM. Since mLSTM does not involve memory mixing, this recurrence relation can be reformulated in a parallel version.

2.3. KAN-xLSTM

The KAN-xLSTM framework substitutes the linear layers in xLSTM with KAN layers. KAN layers parameterize the univariate activation function with learnable B-pline functions, achieving much higher parameter efficiency than traditional linear layers. The final model can be trained in a parallelized way and can converge efficiently through a deep optimization algorithm.

For sLSTM, as Figure 3 shows, the linear layers are considered as follows:

$$
z_t = \sum_{l=1}^{L} \Phi_{z_l} \left(\sum_{p=1}^{P} \phi_{l,p}(x_t, h_{t-1}) \right), \tag{9}
$$

$$
i_t = \exp \left\{ \sum_{l=1}^{L} \Phi_{i_l} \left(\sum_{p=1}^{P} \phi_{l,p}(x_t, h_{t-1}) \right) \right\}, \tag{10}
$$

$$
f_t = \sigma(\tilde{f}_t) \text{ or } \exp(\tilde{f}_t), \quad \tilde{f}_t = \sum_{l=1}^{L} \Phi_{f_l} \left(\sum_{p=1}^{P} \phi_{l,p}(x_t, h_{t-1}) \right), \tag{11}
$$

$$
o_t = \sigma \left\{ \sum_{l=1}^{L} \Phi_{o_l} \left(\sum_{p=1}^{P} \phi_{l,p}(x_t, h_{t-1}) \right) \right\}. \tag{10}
$$

For mLSTM, as Figure 4 shows, the linear layers are given as follows:

$$
q_t = \sum_{l=1}^{L} \Phi_{q_l} \left(\sum_{p=1}^{P} \phi_{l,p}(x_t) \right), \tag{12}
$$

$$
k_t = \frac{1}{\sqrt{d}} \sum_{l=1}^{L} \Phi_{k_l} \left(\sum_{p=1}^{P} \phi_{l,p}(x_t) \right), \tag{13}
$$

$$
v_t = \sum_{l=1}^{L} \Phi_{v_l} \left(\sum_{p=1}^{P} \phi_{l,p}(x_t) \right). \tag{11}
$$

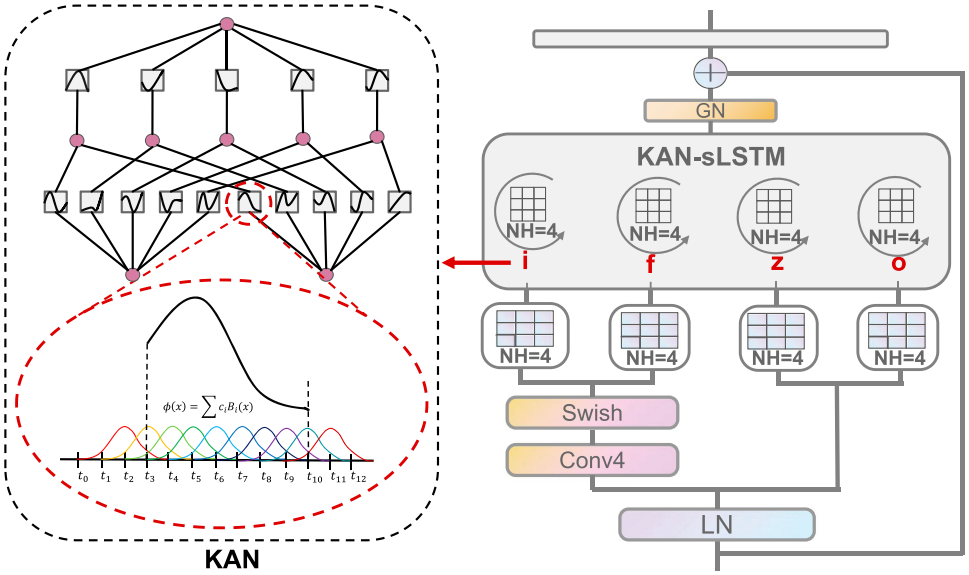

Figure 3. The framework of KAN-sLSTM.

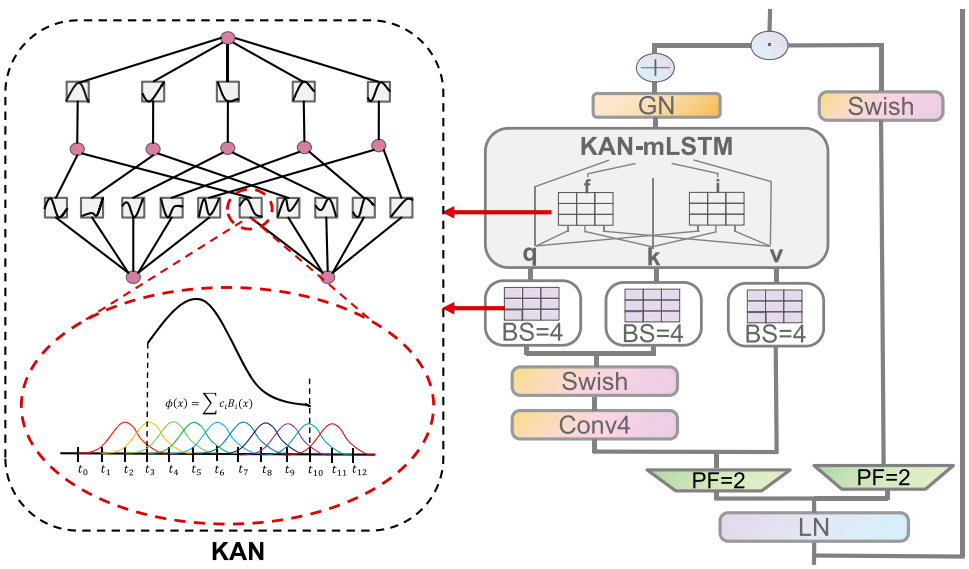

Figure 4. The framework of KAN-mLSTM.

When stacking these mLSTM layers and sLSTM by following the method of Beck et al. (2024), the so-called xLSTM is obtained. After replacing the memory unit of xLSTM with KAN, KANxLSTM is obtained.

The following pseudo-code (Algorithm 1) shows the procedure of KAN-xLSTM model training during pre-training and model fine-tuning.

Considering the randomness of the training of the model, 5 experiments are carried out, and the mean value is taken as the prediction performance. In the next section of the experiment, the effect of window size and step size on RUL prediction accuracy is analyzed, and

then the proposed method is compared with many existing prediction models to comprehensively verify the superiority of the proposed method in the prediction task of residual life of lithium battery.

During the training process, the model's performance is evaluated using cross-validation data to effectively guide the training. The maximum training epochs are set at 100, and early stopping is implemented to halt training promptly. To ensure a fair comparison, all models are trained using the same training protocol. The maximum number of training epochs for each model is set to 100. In addition, we employ an early-stopping strategy to prevent over-fitting and reduce unnecessary computation: training is terminated if the validation loss does not improve for 10 consecutive epochs. This setup ensures that performance differences primarily reflect the capabilities of the models rather than the duration of training.

Algorithm 1 KAN-xLSTM model training procedure

Initialize model parameters Θ (weights for xLSTM and KAN)
Initialize memory states $C_0 = \mathbf{0}_{d\times d}$, $n_0 = \mathbf{0}_{d\times 1}$
for each batch (X, Y) in Dataset **do**
    Let input sequence be $X = \{x_1, x_2, \ldots, x_T\}$
    for $t = 1$ to T **do**
        if block is sLSTM **then**
            $c_t \leftarrow f_t c_{t-1} + i_t z_t$
            $n_t \leftarrow f_t n_{t-1} + i_t$
            $\tilde{h}_t \leftarrow c_t / n_t, \quad h_t \leftarrow o_t \odot \tilde{h}_t$
        else if block is mLSTM **then**
            $C_t \leftarrow f_t C_{t-1} + i_t v_t k_t^\top$
            $n_t \leftarrow f_t n_{t-1} + i_t k_t$
            $\tilde{h}_t \leftarrow C_t q_t / \max\{|n_t^\top q_t|, 1\}$
            $h_t \leftarrow o_t \odot \tilde{h}_t$
        end if
    end for
    Extract final hidden state: $h_{\text{final}} = h_T$
    KAN Layer output:
    $y_{\text{pred}} = \text{Concat}[\phi_1(h_{\text{final}}), \ldots, \phi_L(h_{\text{final}})]$
    Calculate Loss $\mathcal{L}$ and gradients $\nabla_\Theta \mathcal{L}$ via backpropagation
    Update parameters using Adam: $\Theta \leftarrow \Theta - \eta \cdot \text{Adam}(\nabla_\Theta \mathcal{L})$
end for

3. Lithium batteries RUL prediction

3.1. Experimental setting

3.1.1. Introduction to datasets

The dataset used in this paper is from the publicly available NASA Prognostics Center of Excellence (PCoE) Lithium-ion Battery Aging Dataset (Saha & Goebel, 2007). It includes experimental run-to-failure test results of four 18650-sized lithium-ion batteries, numbered

B0005, B0006, B0007, and B0018. This dataset serves as a standard benchmark in RUL prediction research due to its representativeness, accessibility, and comprehensive coverage of battery aging behaviours under consistent experimental protocols.

All tests were conducted at a controlled room temperature of 24°C, with each operational cycle comprising three consecutive procedures: charging, discharging, and Electrochemical Impedance Spectroscopy (EIS) measurements. This design provides multi-sensor measurements (voltage, current, temperature, impedance) for observing the impact of battery aging on both external performance and internal electrochemical parameters. The charging protocol follows a standard CC-CV pattern, starting with a Constant Current (CC) phase at 1.5A until the voltage reaches 4.2V, then switching to a Constant Voltage (CV) phase until the current drops to 20mA. Discharging is carried out at a constant current of 2A until the voltage decreases to a specified end-voltage. To induce accelerated aging, this end-voltage was intentionally set below the manufacturer's recommended lower limit (varying between 2.0V and 2.7V). Each battery's experiment terminates upon meeting the predefined end-of-life (EOL) criterion, defined as a 30% fade in rated capacity from the nominal 2.0 Ah down to 1.4 Ah. The aggregated dataset contains 616 operational procedures across the four batteries.

A key characteristic of this dataset is the unit-to-unit variability exhibited by the batteries. While B0005, B0006, and B0007 each underwent 168 cycles before reaching EOL, B0018 reached its EOL after only 132 cycles. This controlled-yet-realistic variability, coupled with clear degradation trajectories and publicly available data, facilitates reproducible evaluation and comparative analysis of predictive models, making the dataset an appropriate benchmark for validating the proposed KAN-xLSTM framework.

3.1.2. Data preprocessing and spilting

In this paper, the RUL of lithium batteries is predicted using the Capacity feature. In the context of RUL prediction for lithium-ion batteries, the relationship between capacity degradation and remaining useful life is formally defined. Let C_t denote the battery capacity at cycle t, and C_{EOL} be the end-of-life capacity threshold (e.g., 70% of the nominal capacity, or 1.4 Ah). The Remaining Useful Life (RUL) at cycle t is defined as the smallest number of future cycles τ after which the capacity falls below this threshold:

$$\text{RUL}(t) = \inf\{\tau \geq 0 : C_{t+\tau} \leq C_{\text{EOL}}\}.$$

Therefore, the task of predicting RUL is mathematically equivalent to forecasting the future capacity trajectory $\{C_{t+1}, C_{t+2}, \ldots\}$ until it crosses C_{EOL}. Our model directly predicts the capacity values $\widehat{C}_{t+\tau}$ using historical degradation features. The RUL estimate is subsequently derived by identifying the cycle index τ at which the predicted capacity trajectory first satisfies $\widehat{C}_{t+\tau} \leq C_{\text{EOL}}$. This equivalence justifies our use of capacity prediction as the core computational task for RUL estimation, ensuring consistency in model training and evaluation.

Model inputs include Time, Voltage, Current, and Temperature features. A sliding window technique is applied, with the total data amount of a battery denoted as T, and the window size and step as l and m, respectively. For the i-th sample, the input data size is $I \times n$, where n is the number of battery data features. For a battery's data, $\frac{T-1}{m} + 1$ samples can be obtained. For the i-th sample of a battery, the feature data of cycles $m * (i - 1) + 1, m * (i - 1) + 2, \ldots, m * (i - 1) + I$ are used as model inputs, and the Capacity of the $m * (i - 1) + I + 1$ sample is the model output. Normalization is carried out to eliminate the impact of dimension. Three sets of data from batteries 5, 6, 7, and 18 are used as

the training set, and the remaining one set as the test set. The trained model is then used to predict the RUL based on the test set's battery feature data.

3.1.3. Evaluation index

For the quantitative evaluation of the prediction performance of RUL, MAE (Mean Absolute Error), MAPE (Mean Absolute Percentage Error) and RMSE (Root Mean Square Error) are used as the main evaluation indexes, and the calculation formula is as follows:

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\mathrm{y}_i - \tilde{\mathrm{y}}_i\right)^2}, \tag{14}$$

$$\mathrm{MAE} = \frac{1}{N}\sum_{i=1}^{N}\left|\mathrm{y}_i - \tilde{\mathrm{y}}_i\right|, \tag{15}$$

$$\mathrm{MAPE} = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{\mathrm{y}_i - \tilde{\mathrm{y}}_i}{\mathrm{y}_i}\right| \times 100\%, \tag{12}$$

where y_i and $\tilde{\mathrm{y}}_i$ represent the actual Capacity and estimated Capacity of the i-th lithium battery cycle, respectively, and N represents the total number of cycles.

The lower the values of these evaluation metrics, the better the RUL prediction performance, as they are inversely related to it. MAE and MAPE offer distinct perspectives on RUL prediction errors. MAE measures the average absolute difference between predicted and actual values, providing a stable error assessment less affected by outliers, and reflects a model's overall prediction accuracy. MAPE, expressed as a percentage, gauges relative error, making it suitable for comparing datasets of different scales, especially when prediction results have large value ranges, as it better evaluates a model's relative performance.

3.2. Experimental procedure and parameter optimization

3.2.1. Parameter selection

In our RUL prediction experiments, a faster-computing KAN implementation is used with default parameters for its memory units. For xLSTM, this paper used a one-layer mLSTM and a one-layer sLSTM, with an input dimension of 5, 2 attention heads, and a feature dimension of 4 per head.

Sliding window size and step, key to data preprocessing, were systematically evaluated using RMSE. The paper tested window sizes of 2, 4, 10, 20 and steps of 1, 5, 10, 20. To optimize efficiently, this paper first fixed the step at 1 to find the optimal window size, and then optimized the step.

3.2.2. Selection of the window size

As shown in Figure 5, the four battery groups exhibit diverse patterns under varying window sizes. B0005 and B0007 show minimal sensitivity to window size changes, peaking at a window size of 4. Their capacity fluctuations are small, and a moderate window size offers sufficient historical data to capture the battery life-degradation pattern, avoiding the information deficiency of smaller windows. However, performance declines at window sizes of 10 and 20, likely due to overfitting from increased data dimensions and model complexity. Conversely, B0006's prediction error decreases with larger window sizes, with the optimal

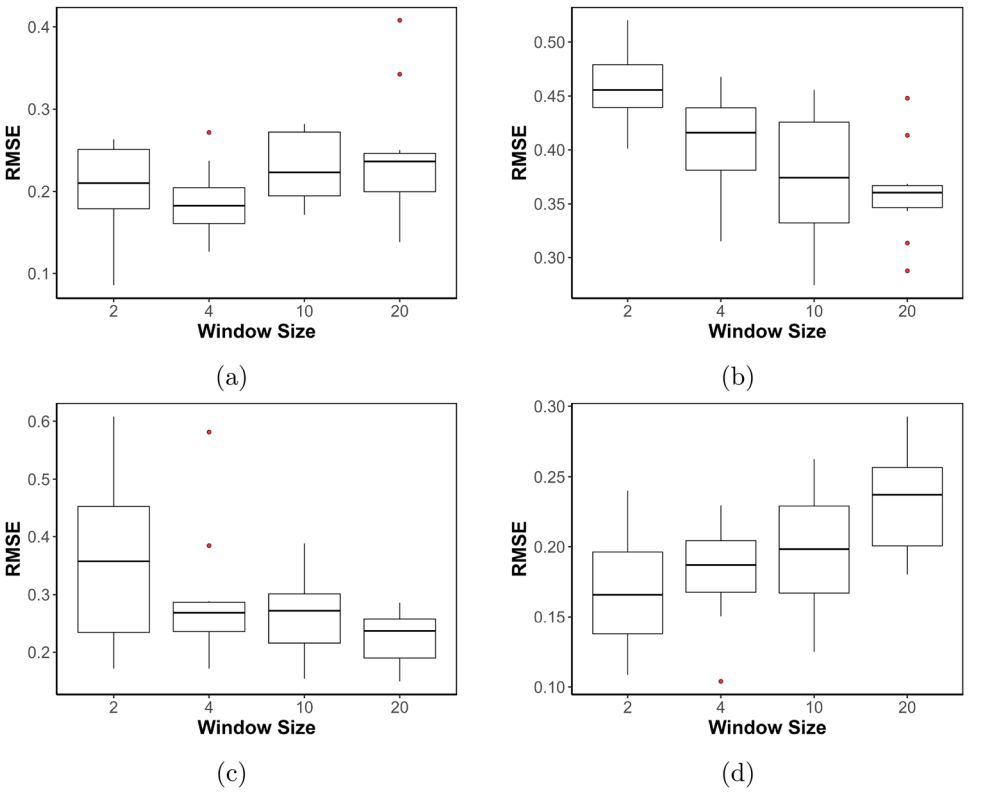

Figure 5. The test RMSE results of KAN-xLSTM with different window sizes, where Figures (a)–(d) corresponding to the B0005, B0006, B0007, B0018, respectively.

size being 20. B0018 is the opposite, with the best performance at a window size of 2. Selecting appropriate window sizes for different batteries balances information volume and model complexity, optimizing prediction performance.

3.2.3. Selection of step size

As depicted in Figure 6, the model's predictive performance is influenced by step sizes, with larger steps leading to increased prediction bias. Battery degradation is governed by long-term charge/discharge patterns and environmental influences. A small step size may inadequately capture these underlying trends, whereas a large step size can oversmooth critical local degradation features. Therefore, to effectively retain temporal detail and enhance prediction accuracy, the step size is fixed at 1 throughout this study.

3.2.4. Hyperparameter sensitivity analysis

Based on the experiments in Section 3.2, the optimal window length and step size that achieve relatively balanced overall performance can be identified. Therefore, in this section, we fix this time-window and step-size configuration as well as the training hyper-parameters, and carry out a systematic sensitivity analysis on the following three core structural hyper-parameters of KAN-xLSTM: 1) the number of B-spline grid points in KAN $\in \{2, 4, 8, 16\}$; 2) the number of attention heads $\in \{1, 2, 4, 8\}$; 3) the xLSTM layer configuration

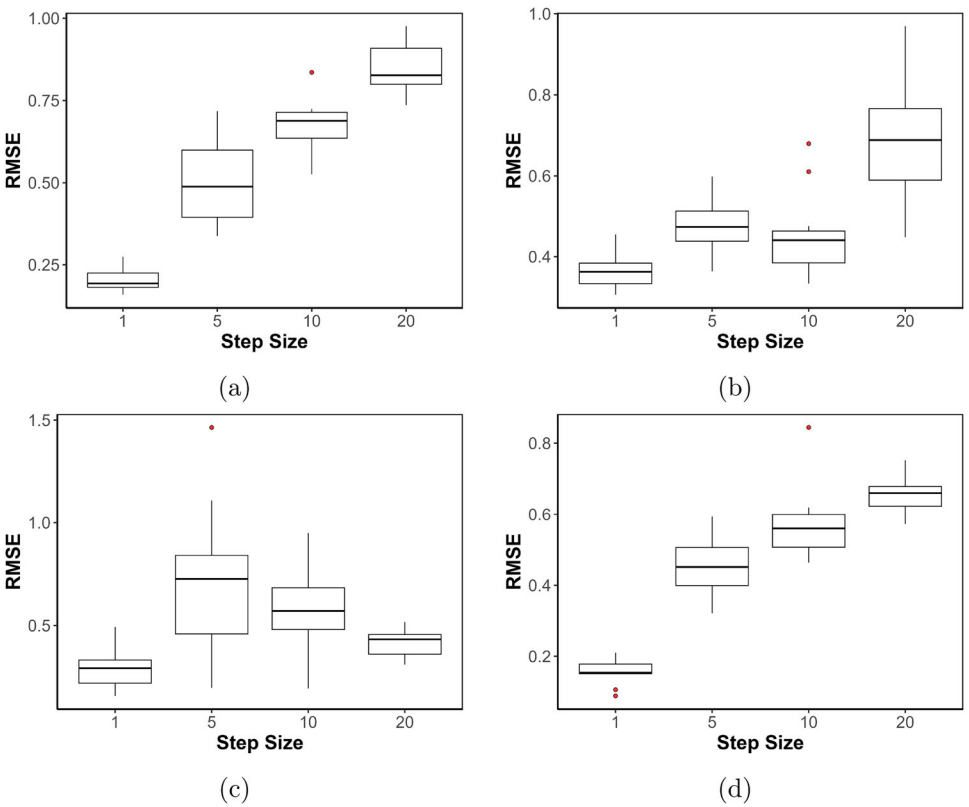

Figure 6. The test RMSE results of KAN-xLSTM with different step sizes, where Figures (a)–(d) corresponding to the B0005, B0006, B0007, B0018, respectively.

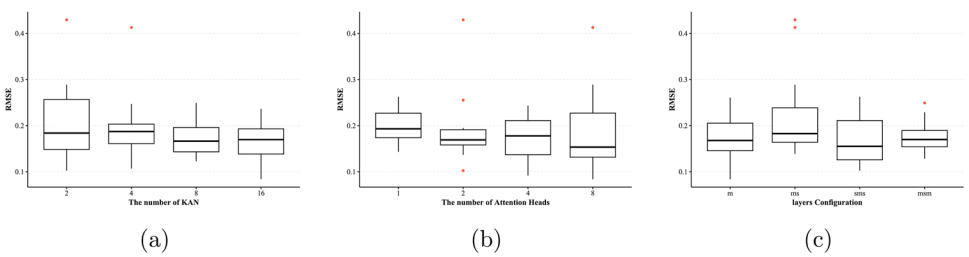

Figure 7. Sensitivity analysis of KAN-xLSTM structural hyper-parameters on RMSE: (a) number of *B*-spline grid points; (b) number of attention heads; (c) layer configuration. Results are obtained from battery B0018.

$\in$ {'m', 'ms', 'sms', 'msm'}, where 's' denotes an sLSTMBlock and 'm' denotes a KANmLSTMBlock (e.g., 'sms' indicates a three-layer stack of s–m–s). For the above $4 \times 4 \times 4 = 64$ combinations, we evaluate RMSE, MAE, and MAPE on the test windows of battery B0018, as shown in Figure 7.

3.3. RUL prediction results

To verify the effectiveness of our model in predicting the remaining useful life (RUL) of lithium batteries, we compare KAN-xLSTM with representative deep-learning baselines,

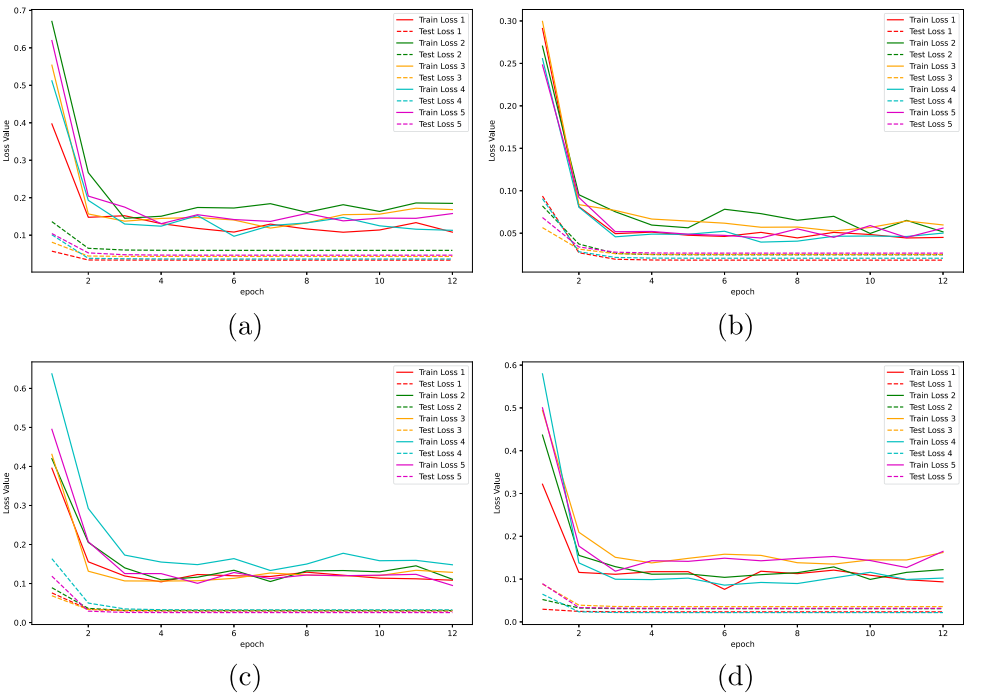

Figure 8. KAN-xLSTM training and testing loss over epochs, where Figures (a)–(d) correspond to the B0005, B0006, B0007, B0018, respectively.

including CNN, LSTM, BiLSTM and xLSTM. All models are configured at comparable parameter levels to ensure a fair comparison and fully demonstrate their predictive capabilities.

As shown in Figure 8, both the training and test losses of KAN-xLSTM on the four battery groups (B0005, B0006, B0007 and B0018) continuously decrease as the number of training epochs increases, indicating that the model gradually captures the degradation characteristics and improves its fitting quality during the learning process. Moreover, the similar downward trends across multiple independent runs demonstrate the model's stable convergence behaviour and good training robustness. Figure 9 further shows that for all four battery groups, the predictions produced by KAN-xLSTM are closest to the true values. Among the baseline models, thanks to its enhanced memory and gating design, xLSTM generally outperforms LSTM/BiLSTM/CNN, while KAN-xLSTM further improves upon xLSTM by incorporating KAN-based nonlinear representations.

The quantitative results in Tables 1–4 corroborate these observations. Specifically, KAN-xLSTM achieves the lowest RMSE and MAE across all four battery groups, indicating consistent improvement in absolute capacity prediction accuracy. In addition, it obtains the best MAPE on B0007 and B0018, and remains competitive on B0005 and B0006.

We note that MAPE reflects the relative prediction error, which is often amplified when the true RUL value is small and close to end-of-life due to the diminishing denominator. This sensitivity may explain why MAPE is slightly higher for KAN-xLSTM in some cases (e.g., B0005 and B0006), even though its absolute error (RMSE/MAE) is lower. More importantly, RUL prediction primarily focuses on the accuracy of later-stage predictions, especially

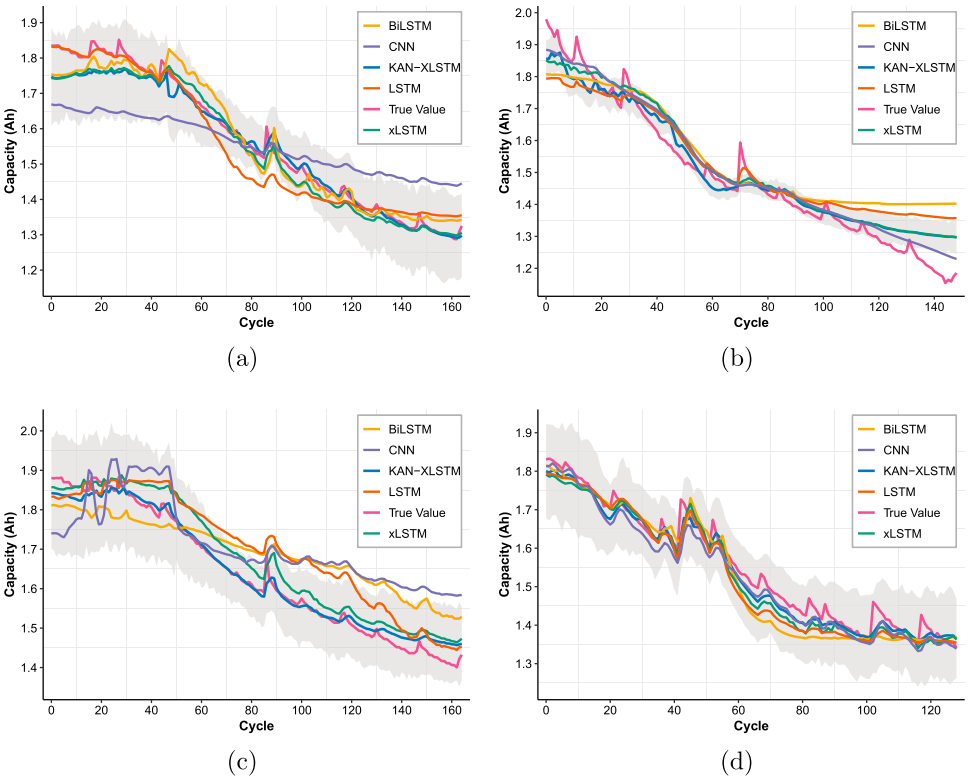

Figure 9. RUL prediction results with uncertainty quantification. (a)–(d) show the predicted RUL curves (solid lines) along with their 95% confidence intervals (shaded regions) obtained via MC Dropout for batteries B0005, B0006, B0007 and B0018, respectively.

Table 1. B0005 prediction results.

Model	RMSE	MAE	MAPE
CNN	0.215	0.184	58.980
LSTM	0.546	0.480	24.159
BiLSTM	0.220	0.183	25.539
xLSTM	0.212	0.166	**16.685**
KAN-xLSTM	**0.194**	**0.158**	17.568

Table 2. B0006 prediction results.

Model	RMSE	MAE	MAPE
CNN	0.517	0.393	2.103
LSTM	0.983	0.844	**1.288**
BiLSTM	0.452	0.356	5.564
xLSTM	0.366	0.307	1.991
KAN-xLSTM	**0.363**	**0.295**	2.192

as the battery approaches its end-of-life threshold. As shown in the prediction curves, KAN-xLSTM demonstrates superior alignment with the true degradation trajectory in these critical late-stage cycles, which is more relevant to practical maintenance decisions than a single percentage metric like MAPE. Therefore, while MAPE provides a supplementary perspective, the consistently superior performance in RMSE and MAE confirms that KAN-xLSTM offers

Table 3. B0007 prediction results.

Model	RMSE	MAE	MAPE
CNN	0.548	0.454	1.827
LSTM	0.620	0.553	6.269
BiLSTM	0.601	0.549	5.542
xLSTM	0.392	0.255	2.861
KAN-xLSTM	**0.263**	**0.223**	**1.246**

Table 4. B0018 prediction results.

Model	RMSE	MAE	MAPE
CNN	0.439	0.385	0.823
LSTM	0.231	0.187	1.005
BiLSTM	0.318	0.250	2.468
xLSTM	0.189	0.156	0.492
KAN-xLSTM	**0.153**	**0.119**	**0.433**

more reliable and accurate predictions for battery health management. Overall, our model demonstrates better comprehensive performance compared to the baseline models.

Beyond point estimates, reliable RUL prediction should also account for prediction uncertainty to support risk-informed maintenance decisions. In the following subsection, we therefore introduce a Monte Carlo Dropout-based uncertainty quantification framework, which enables KAN-xLSTM to provide not only accurate point predictions but also well-calibrated confidence intervals.

To complement the point predictions presented above, we equip KAN-xLSTM with Monte Carlo (MC) Dropout, a practical Bayesian approximation that quantifies prediction uncertainty through repeated stochastic forward passes during inference. By keeping dropout active at test time, the random masking of neurons can be interpreted as sampling from the approximate posterior distribution of the network weights, yielding an ensemble of predictions from which uncertainty can be estimated.

Concretely, for each test sample we perform $N_{\mathrm{MC}} = 100$ forward passes with different dropout masks, obtaining a set of predictions $\{\hat{y}^{(k)}\}_{k=1}^{N_{\mathrm{MC}}}$. The predictive mean μ and standard deviation σ are computed as

$$\mu = \frac{1}{N_{\mathrm{MC}}} \sum_{k=1}^{N_{\mathrm{MC}}} \hat{y}^{(k)}, \quad \sigma = \sqrt{\frac{1}{N_{\mathrm{MC}} - 1} \sum_{k=1}^{N_{\mathrm{MC}}} \left(\hat{y}^{(k)} - \mu\right)^2}.$$

The 95% confidence interval (CI) is then constructed as

$$\mathrm{CI}_{95\%} = [\, \mu - 1.96\sigma,\ \mu + 1.96\sigma \,].$$

All features are standardized during training, so μ, σ, and the CI endpoints are computed in the normalized space and then mapped back to the original physical scale for the target variable only.

Figure 9 displays the 95% confidence intervals obtained by this procedure for each battery. To evaluate the quality of the uncertainty estimates, we report two additional metrics alongside the conventional accuracy measures: (1) *CI coverage* – the proportion of true values that fall within the predicted 95% CI, indicating calibration; and (2) *average CI width* – the mean width of all CIs, reflecting the tightness of the prediction intervals.

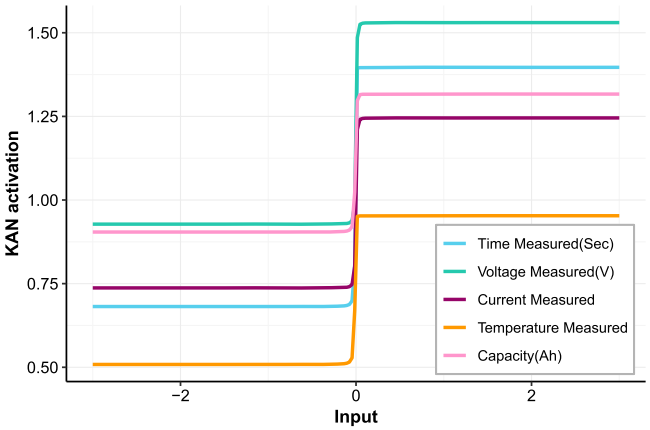

Figure 10. The *B*-spline activation functions of the KAN layers on battery B0018.

3.4. Interpretability analysis

3.4.1. Visual analysis of KAN nonlinear activations

To elucidate the internal mechanism of the proposed FasterKAN-xLSTM, we extract the univariate activation functions of the KAN modules in the first KANmLSTM layer. For each input feature, we vary its normalized value along the horizontal axis while fixing the other features to zero, and record the scalar output after passing through the KAN basis functions. The resulting curve can be regarded as the 'univariate response function' of the model to each input variable.

As shown in Figure 10, the activation functions for the five input variables exhibit a consistent threshold-type nonlinearity: the KAN output stays at a low plateau for inputs below zero, transitions sharply within a narrow region near zero, and then rises to a higher plateau that gradually saturates as the input increases further.

Thus, the KAN essentially partitions each input variable into 'low/high' intervals and performs strong nonlinear amplification around the threshold, thereby controlling memory updates through the gating mechanism of KAN-mLSTM. This indicates that the model relies more on whether a variable falls within a certain range or side of the threshold, rather than utilizing its continuous values linearly.

3.4.2. Gate-wise feature importance visualization

Figure 10 shows the normalized importance distribution of input features across different gating units in the proposed KAN-xLSTM obtained from battery B0018. The importance is obtained by averaging the absolute values of each gate's weights over the hidden-unit dimension and normalizing them to [0, 1]. A higher value indicates that the corresponding feature exerts a stronger influence on the gating decision.

For the sLSTM (Figure 11), it can be observed that all three gates exhibit high sensitivity to the voltage feature. The output gate gives voltage the highest relative importance (1.0), while the forget gate and input gate assign importance scores of 0.84 and 0.81 to voltage, respectively. This indicates that the model heavily relies on voltage information when updating and exposing the hidden state, which aligns with the physical fact that voltage is highly sensitive to the instantaneous state of a battery. The input gate also shows noticeably higher weights for current (0.90) and temperature (0.87), whereas the weight for capacity is relatively low

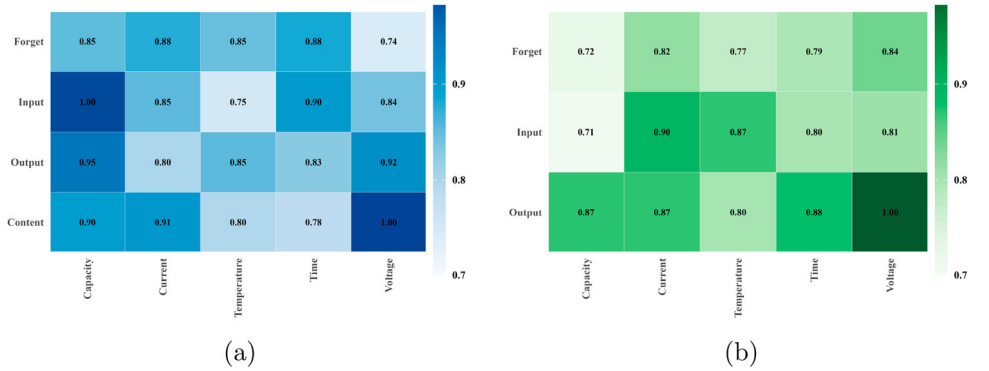

Figure 11. Gate-wise feature importance visualization. (a) Normalized importance of input features for the Forget, Input, and Output gates in the baseline sLSTM. (b) Normalized importance for the Forget, Input, Output gates and the Content pathway in the proposed mLSTM. Darker color denotes higher importance.

(0.71). This suggests that the baseline sLSTM focuses primarily on load and temperature conditions when admitting new information, while the signal of long-term capacity degradation is attenuated.

In contrast, the importance distribution across gates and features changes markedly in the mLSTM module (Figure 11). First, the importance of capacity and time increases significantly for the input gate, with capacity reaching 1.00 and time reaching 0.90, whereas the importance of temperature decreases. This implies that KAN-xLSTM depends more on slow variables such as capacity and time–which reflect long-term degradation–when deciding to 'write' new memories, rather than over-relying on instantaneous operating features. Second, the forget gate shows a more balanced weight distribution across capacity, current, temperature, and time, while its reliance on voltage drops noticeably (0.74). This indicates that the model takes into account comprehensive information about overall operating conditions and degradation levels when forgetting past states.

Overall, KAN-xLSTM allocates instantaneous operating features (voltage, current) and long-term degradation features (capacity, time) more reasonably across different gates and pathways. This results in an internal decision-making process that is more consistent with the physical mechanism of battery degradation, which also partly explains its superior prediction performance.

3.5. Supplementary experiment

To verify the generalization capability of the KAN-xLSTM model, this study conducts supplementary validation using a lithium-ion battery dataset (He et al., 2011) from the University of Maryland. The experiment selects CS2-type batteries, which are cycled at a constant current of 1C, with battery models CS2_35, CS2_36, CS2_37, and CS2_38. In terms of parameter settings, the window size is set to 30 and the number of epochs to 200 based on grid search, while other parameters remain consistent with previous experiments. The baseline models are also configured identically. Figure 12 displays the predicted curve of battery capacity for five models. As shown in Tables 5–8, KAN-xLSTM achieves lower RMSE, MAE, and MAPE compared to the other baseline models, demonstrating its better performance on this dataset and thus proving its promising generalization capability.

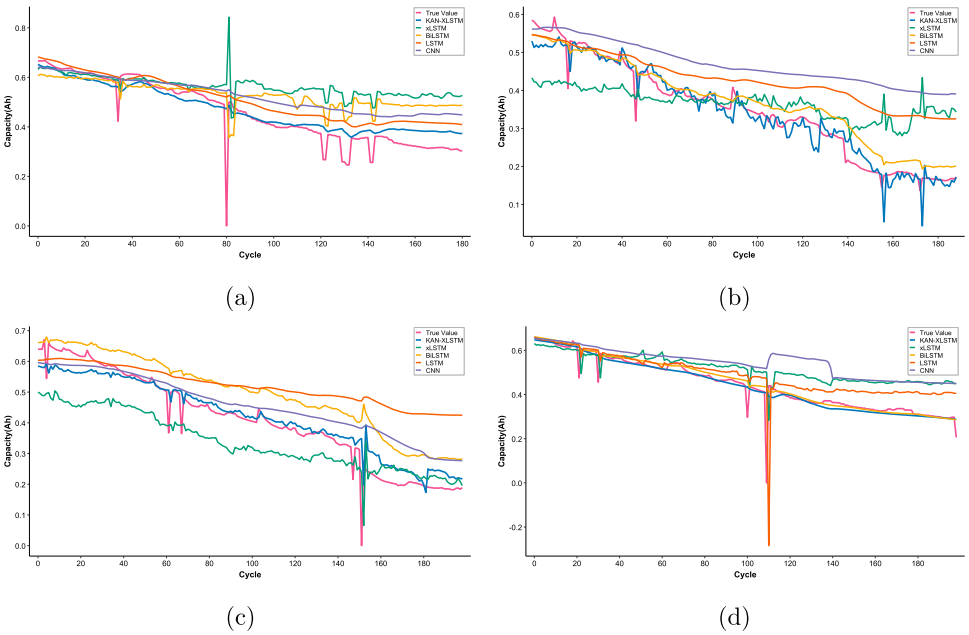

Figure 12. Capacity prediction results on the University of Maryland battery dataset: (a) CS2_35, (b) CS2_36, (c) CS2_37, and (d) CS2_38.

Table 5. CS2_35 prediction results.

Model	RMSE	MAE	MAPE (%)
CNN	0.098	0.074	30.103
LSTM	0.076	0.054	28.753
BiLSTM	0.115	0.091	29.729
xLSTM	0.154	0.113	31.492
KAN-xLSTM	**0.056**	**0.036**	**26.090**

Table 6. CS2_36 prediction results.

Model	RMSE	MAE	MAPE (%)
CNN	0.142	0.122	51.212
LSTM	0.102	0.085	36.179
BiLSTM	0.038	0.031	11.816
xLSTM	0.098	0.080	31.333
KAN-xLSTM	**0.032**	**0.022**	**7.376**

4. Conclusion

This paper proposes a hybrid KAN-xLSTM architecture for improved prediction of lithium battery RUL. The method exploits two main ideas: KAN replaces traditional linear weights with *B*-spline–parameterized univariate functions to enhance flexibility and high-dimensional function approximation, while xLSTM introduces exponential gating and covariance updates to better capture long-term dependencies. Experiments on the NASA lithium battery aging dataset show that KAN-xLSTM outperforms conventional models in prediction accuracy, especially under large capacity fluctuations. The KAN-xLSTM achieves the lowest RMSE and MAE across all battery groups, with particularly strong gains for

Table 7. CS2_37 prediction results.

Model	RMSE	MAE	MAPE (%)
CNN	0.077	0.060	18.902
LSTM	0.147	0.120	23.323
BiLSTM	0.097	0.088	20.229
xLSTM	0.099	0.088	17.259
KAN-xLSTM	**0.048**	**0.033**	**11.769**

Table 8. CS2_38 prediction results.

Model	RMSE	MAE	MAPE (%)
CNN	0.077	0.060	18.902
LSTM	0.147	0.120	23.323
BiLSTM	0.097	0.088	20.229
xLSTM	0.099	0.088	17.259
KAN-xLSTM	**0.048**	**0.033**	**11.769**

batteries with large capacity variations (e.g., B0018), confirming its robustness to nonlinear, multi-timescale degradation. Gate-wise feature-importance visualizations indicate that the model balances instantaneous operating features (e.g., voltage, current) and long-term degradation indicators (e.g., capacity, time) in a way consistent with battery physics, offering a principled explanation for its improved accuracy. Combining KAN's parameter-efficient function approximation with xLSTM's parallel structure also shortens training time while maintaining high accuracy, making real-time or near-real-time RUL estimation feasible.

For deployment in resource-constrained settings, further lightweight optimization–such as pruning, quantization, or distillation–is desirable to reduce memory and computation. The modular design of KAN-xLSTM supports such adaptations, though trade-offs among model complexity, inference speed, and maintenance costs require case-specific study. Overall, the results demonstrate the engineering potential of this hybrid architecture for battery health management and its applicability to other complex degradation analysis tasks. Future work will emphasize model compression, real-time embedded implementation, and cost-benefit analysis in industrial operation and maintenance, to bridge theoretical innovation and practical deployment.

Acknowledgments

The authors thank the anonymous referees and the editor for their useful comments and suggestions on an earlier version of this manuscript, which resulted in this improved version.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Funding

This work was supported in part by the Sichuan Science and Technology Program (grant number 2024ZYD0135), in part by the National Natural Science Foundation of China (grant numbers 12071372, 12201395 and 11901134).

ORCID

Fode Zhang http://orcid.org/0000-0002-4733-9752

References

Ba, J., Hinton, G. E., Mnih, V., Leibo, J. Z., & Ionescu, C. (2016). Using fast weights to attend to the recent past. In *Advances in Neural Information Processing Systems* (Vol. 29, pp. 4338–4346). Neural Information Processing Systems Foundation, Inc.

Beck, M., Pöppel, K., Spanring, M., Auer, A., Prudnikova, O., Kopp, M., Klambauer, G., Brandstetter, J., & Hochreiter, S. (2024). xLSTM: Extended long short-term memory. In *Advances in Neural Information Processing Systems* (Vol. 37, pp. 107547–107603). Neural Information Processing Systems Foundation, Inc.

Bu, Z., Long, B., Liu, Z., Wu, K., Geng, H., & Cheng, Y. (2025). Multivariate adaptive Brownian motion-particle filter framework for remaining useful life prediction of nonlinear and state-noise coupled degradation process. *Reliability Engineering & System Safety*, *264*,111356. https://doi.org/10.1016/j.ress.2025.111356

Cai, X., Li, N., & Xie, M. (2024, April). RUL prediction for two-phase degrading systems considering physical damage observations. *Reliability Engineering & System Safety*, *244*,109926. https://doi.org/10.1016/j.ress.2024.109926

Cui, X., Chen, Z., Lan, J., & Dong, M. (2021). An online state of health estimation method for lithium-ion battery based on ICA and TPA-LSTM. In *2021 IEEE Industrial Electronics and Applications Conference (IEACON)* (pp. 130–135). IEEE.

Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, *2*(4), 303–314. https://doi.org/10.1007/BF02551274

Dayan, P., & Willshaw, D. J. (1991). Optimising synaptic learning rules in linear associative memories. *Biological Cybernetics*, *65*(4), 253–265. https://doi.org/10.1007/BF00206223

Gao, K., Wu, D., Zhang, S., Peng, R., & Wu, S. (2025). The state-of-the-art development and new challenges: Operations management of metro systems. *ICCK Transactions on Systems Safety and Reliability*, *1*(1), 4–20. https://doi.org/10.62762/TSSR.2025.246708

Genet, R., & Inzirillo, H. (2025). Tkan: Temporal Kolmogorov-Arnold networks. https://arxiv.org/abs/2405.07344

Greff, K., Srivastava, R. K., Koutník, J., Steunebrink, B. R., & Schmidhuber, J. (2016). LSTM: A search space odyssey. *IEEE Transactions on Neural Networks and Learning Systems*, *28*(10), 2222–2232. https://doi.org/10.1109/TNNLS.2016.2582924

He, W., Williard, N., Osterman, M., & Pecht, M. (2011). *Lithium-ion Battery Aging Dataset*. Center for Advanced Life Cycle Engineering, University of Maryland.

Hess, A., Calvello, G., & Frith, P (2005, March). Challenges, issues, and lessons learned chasing the 'Big P'. Real predictive prognostics. Part 1. In *2005 IEEE Aerospace Conference* (pp. 3610–3619). IEEE.

Hochreiter, S., & Schmidhuber, J. (1997, November). Long short-term memory. *Neural Computation*, *9*(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Hornik, K., Stinchcombe, M., & White, H. (1989). Multilayer feedforward networks are universal approximators. *Neural Networks*, *2*(5), 359–366. https://doi.org/10.1016/0893-6080(89)90020-8

Kolmogorov, A. N (1961). On the representation of continuous functions of several variables by superpositions of continuous functions of a smaller number of variables. In *Collected works* (pp. 25–46). American Mathematical Society.

Krotov, D., & Hopfield, J. (2018). Dense associative memory is robust to adversarial inputs. *Neural Computation*, *30*(12), 3151–3167. https://doi.org/10.1162/neco_a_01143

Krotov, D., & Hopfield, J. J. (2016). Dense associative memory for pattern recognition. In *Advances in Neural Information Processing Systems* (Vol. 29). Neural Information Processing Systems Foundation, Inc.

Lai, C., Baraldi, P., & Zio, E. (2025, July). Gradient-enhanced physics-informed long short-term memory networks for stable and accurate prediction of the RUL of electronic components. *Reliability Engineering & System Safety*, *265*,111485. https://doi.org/10.1016/j.ress.2025.111485

Li, N., Wang, M., Lei, Y., Yang, B., Li, X., & Si, X. (2025). Remaining useful life prediction of lithium-ion battery with nonparametric degradation modeling and incomplete data. *Reliability Engineering & System Safety*, *256*,110721. https://doi.org/10.1016/j.ress.2024.110721

Li, S., Fang, H., & Shi, B. (2021). Remaining useful life estimation of lithium-ion battery based on interacting multiple model particle filter and support vector regression. *Reliability Engineering & System Safety*, *210*,107542. https://doi.org/10.1016/j.ress.2021.107542

Li, Y., Xiong, B., Vilathgamuwa, D. M., Wei, Z., Xie, C., & Zou, C. (2021, January). Constrained ensemble Kalman filter for distributed electrochemical state estimation of lithium-ion batteries. *IEEE Transactions on Industrial Informatics*, *17*(1), 240–250. https://doi.org/10.1109/TII.9424

Liu, J., Hou, B., Lu, M., & Wang, D. (2024). Box-Cox transformation based state-space modeling as a unified prognostic framework for degradation linearization and RUL prediction enhancement. *Reliability Engineering & System Safety*, *244*, 109952. https://doi.org/10.1016/j.ress.2024.109952

Liu, K., Hu, X., Wei, Z., Li, Y., & Jiang, Y. (2019, December). Modified Gaussian process regression models for cyclic capacity prediction of lithium-ion batteries. *IEEE Transactions on Transportation Electrification*, *5*(4), 1225–1236. https://doi.org/10.1109/TTE.6687316

Liu, K., Shang, Y., Ouyang, Q., & Widanage, W. D. (2021, April). A data-driven approach with uncertainty quantification for predicting future capacities and remaining useful life of lithium-ion battery. *IEEE Transactions on Industrial Electronics*, *68*(4), 3170–3180. https://doi.org/10.1109/TIE.41

Liu, Z., Wang, Y., Vaidya, S., Ruehle, F., Halverson, J., Soljačić, M., Hou, T. Y., & Tegmark, M. (2024). Kan: Kolmogorov-Arnold networks. Preprint. arXiv:2404.19756

Milakov, M., & Gimelshein, N. (2018). Online normalizer calculation for softmax. Preprint. arXiv:1805.02867

Niu, H., Zeng, J., Shi, H., Zhang, X., Liang, J., & Shi, G. (2025). Remaining useful life prediction for multi-component systems with stochastic correlation based on auxiliary particle filter. *Reliability Engineering & System Safety*, *264*,111357. https://doi.org/10.1016/j.ress.2025.111357

Peiseler, L., Schenker, V., Schatzmann, K., Pfister, S., Wood, V., & Schmidt, T. (2024, May). Carbon footprint distributions of lithium-ion batteries and their materials. Preprint.

Perumal, T., Mustapha, N., Mohamed, R., & Shiri, F. M. (2024). A comprehensive overview and comparative analysis on deep learning models. *Journal on Artificial Intelligence*, *6*(1), 301–360. https://doi.org/10.32604/jai.2024.054314

Ramsauer, H., Schäfl, B., Lehner, J., Seidl, P., Widrich, M., Adler, T., Gruber, L., Holzleitner, M., Pavlović, M., Sandve, G. K., Greiff, V., Kreil, D., Kopp, M., Klambauer, G., Brandstetter, J., & Hochreiter, S. (2021). Hopfield networks is all you need. Preprint. https://arxiv.org/abs/2008.02217.

Ren, L., Zhao, L., Hong, S., Zhao, S., Wang, H., & Zhang, L. (2018). Remaining useful life prediction for lithium-ion battery: A deep learning approach. *IEEE Access*, *6*, 50587–50598. https://doi.org/10.1109/ACCESS.2018.2858856

Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, *65*(6), 386–408. https://doi.org/10.1037/h0042519

Saha, B., & Goebel, K. (2007). *Battery Data Set*. NASA Ames Research Center, NASA Prognostics Data Repository.

Schlag, I., Irie, K., & Schmidhuber, J. (2021). Linear transformers are secretly fast weight programmers. In *International Conference on Machine Learning* (pp. 9355–9366). Proceedings of Machine Learning Research.

Schmidhuber, J. (1992). Learning to control fast-weight memories: An alternative to dynamic recurrent networks. *Neural Computation*, *4*(1), 131–139. https://doi.org/10.1162/neco.1992.4.1.131

Sejnowski, T. J. (1977). Storing covariance with nonlinearly interacting neurons. *Journal of Mathematical Biology*, *4*(4), 303–321. https://doi.org/10.1007/BF00275079

Shu, Q., Zhang, F., Shen, L., & Ng, H. K. T. (2024, January). RUL prediction with cross-domain adaptation based on reproducing kernel Hilbert space. *IEEE Transactions on Reliability*, *74*(3), 3871–3883. https://doi.org/10.1109/TR.2024.3488792

Sun, Y., Dong, L., Huang, S., Ma, S., Xia, Y., Xue, J., Wang, J., & Wei, F. (2023). Retentive network: A successor to transformer for large language models. Preprint. arXiv:2307.08621

Vaca-Rubio, C. J., Blanco, L., Pereira, R., & Caus, M. (2024). Kolmogorov-Arnold networks (KANs) for time series analysis. Preprint. arXiv:2405.08790

Wang, F. K., Amogne, Z. E., Chou, J. H., & Tseng, C. (2022). Online remaining useful life prediction of lithium-ion batteries using bidirectional long short-term memory with attention mechanism. *Energy*, *254*,124344. https://doi.org/10.1016/j.energy.2022.124344

Wang, H., Ma, X., & Zhao, Y. (2019). An improved Wiener process model with adaptive drift and diffusion for online remaining useful life prediction. *Mechanical Systems and Signal Processing*, *127*, 370–387. https://doi.org/10.1016/j.ymssp.2019.03.019

Wang, J., Zhang, F., Ng, H. K. T., & Shi, Y. (2024). Remaining useful life prediction via information enhanced domain adversarial generalization. *IEEE Transactions on Reliability*, *74*(2), 2837–2850. https://doi.org/10.1109/TR.2024.3441592

Xiang, S., Qin, Y., Luo, J., Pu, H., & Tang, B. (2021). Multicellular LSTM-based deep learning model for aero-engine remaining useful life prediction. *Reliability Engineering & System Safety*, *216*,107927. https://doi.org/10.1016/j.ress.2021.107927

Xiang, S., Qin, Y., Zhu, C., Wang, Y., & Chen, H. (2020). Long short-term memory neural network with weight amplification and its application into gear remaining useful life prediction. *Engineering Applications of Artificial Intelligence*, *91*,103587. https://doi.org/10.1016/j.engappai.2020.103587

Xu, A., Fang, G., Zhuang, L., & Gu, C. (2025, September). A multivariate student-t process model for dependent tail-weighted degradation data. *IISE Transactions*, *57*(9), 1071–1087. https://doi.org/10.1080/24725854.2024.2389538

Zhang, F., Ng, H. K. T., & Shen, L. (2023, January). Robust estimation and selection for degradation modeling with inhomogeneous increments. *IEEE Transactions on Reliability*, *73*, 560–575.

Zhang, S., Zhai, Q., & Li, Y. (2023). Degradation modeling and RUL prediction with wiener process considering measurable and unobservable external impacts. *Reliability Engineering & System Safety*, *231*,109021. https://doi.org/10.1016/j.ress.2022.109021

Zhu, R., Hu, J., & Peng, W. (2025). Bayesian calibrated physics-informed neural networks for second-life battery SOH estimation. *Reliability Engineering & System Safety*, *264*,111432. https://doi.org/10.1016/j.ress.2025.111432

Zhu, R., Peng, W., Ye, Z. S., & Xie, M. (2025, January). Collaborative prognostics of lithium-ion batteries using federated learning with dynamic weighting and attention mechanism. *IEEE Transactions on Industrial Electronics*, *72*(1), 980–991. https://doi.org/10.1109/TIE.2024.3387115

Zhu, Y., Zhang, F., Chen, W., Cheng, Z., & Shen, L. (2026). Order-preserving kernel contrastive learning with applications to cross-domain RUL prediction. *IEEE Transactions on Reliability*, *75*, 97–111. https://doi.org/10.1109/TR.2025.3635402