Energy and AI 24 (2026) 100694

Contents lists available at ScienceDirect

Energy and AI

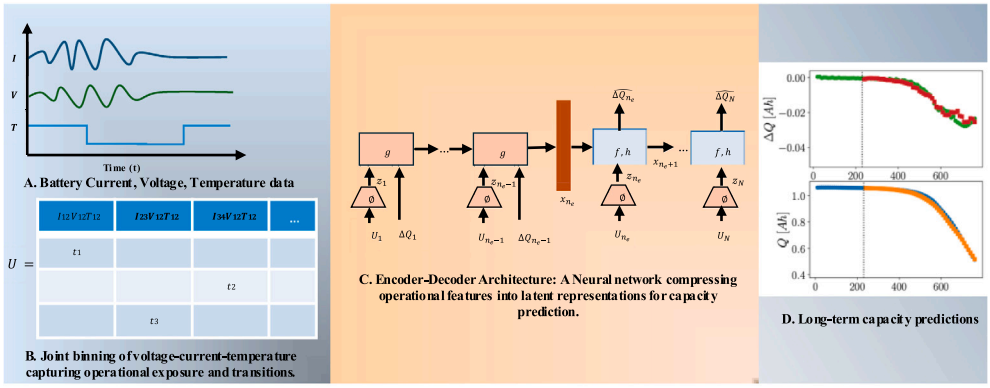
journal homepage: www.elsevier.com/locate/egyai

Forecasting battery capacity with dynamic transitions in operational conditions

Tushar Desai *, Riccardo M.G. Ferrari

Delft Center for Systems and Control, Delft University of Technology, Mekelweg 2, Delft, 2628 CD, South Holland, Netherlands

GRAPHICAL ABSTRACT



HIGHLIGHTS

- Developed transition-aware encoding for battery operational sequences.
- Validated on encoder–decoder network for long-term capacity forecasting.
- Reduced worst-case capacity prediction errors by over 50%.
- Achieved 94.3% storage reduction while preserving temporal dynamics.

ARTICLE INFO

Keywords:
Battery
Degradation
Forecasting
Machine learning

ABSTRACT

Accurate battery capacity forecasting is crucial for ensuring safe operation and effective maintenance scheduling. However, capacity prediction remains challenging due to the complex, nonlinear degradation processes influenced by diverse operational conditions and usage patterns. Existing operational condition analysis methods either treat voltage, current, and temperature independently, losing cross-variable coupling effects, or aggregate exposure durations without preserving temporal ordering, discarding transition dynamics that influence degradation pathways. This work addresses both limitations through a transition-aware encoding method that discretizes measurements into joint operational bins, tracking the sequence of transitions between bins and preserving both coupled effects and temporal dynamics. An encoder–decoder neural network processes these compact representations to generate capacity forecasts over extended horizons. Based on experimental data from lithium–iron–phosphate (LFP) cells undergoing nonlinear degradation, the proposed transition-aware

* Corresponding author.
E-mail addresses: t.k.desai@tudelft.nl (T. Desai), r.ferrari@tudelft.nl (R.M.G. Ferrari).

https://doi.org/10.1016/j.egyai.2026.100694
Received 13 September 2025; Received in revised form 13 December 2025; Accepted 7 February 2026
Available online 14 February 2026

encoding forecasts absolute capacity with a mean absolute percentage error of 1.68% and captures cycle-to-cycle capacity variation to within 0.16%, while simultaneously compressing raw time-series data by 94.3%. Compared to methods that discard temporal ordering or treat measurements independently, the proposed approach reduces worst-case capacity prediction errors by more than 50%.

1. Introduction

Battery systems, particularly lithium-ion batteries (LIBs), have become integral to numerous critical applications, including electric vehicles, grid-scale renewable energy storage, and portable electronic devices. This surge in application is due to their high energy density, decreasing costs, and prolonged cycle life [1]. At the same time, large-scale electrification is shifting battery management from single-pack supervision to fleet-level health monitoring, where capacity prognostics directly support warranty decisions, maintenance planning, and second-life valuation under strict on-board computation and memory constraints [2]. As society transitions toward electrification and sustainable energy systems, the reliability and performance of battery systems become increasingly crucial to achieving both economic and environmental objectives.

However, battery degradation remains an inevitable reality. Through complex electrochemical processes, including solid-electrolyte interphase (SEI) formation, lithium plating, electrode particle cracking, and electrolyte decomposition, batteries progressively lose their energy storage capacity [3]. This capacity fade not only reduces operational performance but can also precipitate safety hazards through thermal runaway or sudden failure. Consequently, accurate capacity forecasting has become essential for ensuring safe operation, enabling predictive maintenance, optimizing replacement schedules, and minimizing operational costs across diverse applications.

Forecasting battery capacity accurately remains a fundamentally challenging task due to several factors. The degradation process involves multiple coupled nonlinear mechanisms where each one influences others through intricate feedback loops [3]. Manufacturing tolerances introduce cell-to-cell variations even within single production batches, causing divergent aging trajectories under identical conditions [4]. Real-world operating conditions vary widely; different charge–discharge rates, temperature exposures, and usage patterns create an expansive parameter space that challenges conventional modeling approaches [5]. In addition, field data are often irregular, partially observed (e.g., partial charges), and affected by sensor noise and data-quality issues, which can reduce the transferability of laboratory-trained models [6].

Existing capacity forecasting methods can be broadly categorized into physics-based, equivalent circuit models (ECM), and data-driven approaches, each with distinct advantages and limitations. Physics-based models, particularly electrochemical models like the pseudo-two-dimensional (P2D) framework, offer detailed insights by explicitly modeling degradation processes such as SEI growth and lithium plating [7]. While these models offer interpretability and physical understanding, they incur high computational costs due to the solution of partial differential equations and require cell-specific parameters that are often difficult to obtain in practice [8]. Equivalent circuit models represent batteries through networks of resistors and capacitors, providing computational efficiency by capturing macroscopic electrical behavior without detailed electrochemical modeling. However, ECM parameters vary nonlinearly with state of charge (SOC), temperature, and current, requiring frequent recalibration to maintain accuracy as batteries age [9]. Although recent efforts have addressed specific operating conditions, such as low-temperature accuracy through multiple-feature electrochemical–thermal coupling with real-time coefficient correction [10], the empirical nature of ECMs fundamentally limits their ability to extrapolate beyond calibration conditions or capture long-term degradation mechanisms [11].

On the other hand, data-driven approaches have recently gained significant traction for battery capacity forecasting, due to advances in machine learning (ML) techniques and availability of open-source datasets [12]. ML-based methods can be broadly categorized into feature-based and end-to-end approaches. Feature-based methods extract handcrafted features from battery data and map them to degradation outcomes using ML algorithms. Examples include electrochemical transition features derived from differential voltage (dV/dQ) analysis, such as peak positions and magnitudes that indicate phase changes in electrode materials [13], and capacity fade indicators extracted from specific charging segments such as constant-current or constant-voltage phases [14]. Beyond single-output regression, multi-task learning has been shown to jointly forecast capacity fade and power/impedance-related degradation trajectories (including knee points), improving robustness versus single-task baselines [15]. Li et al. [16] further demonstrated that early-life features can enable lifetime prediction even under widely varying usage conditions, highlighting both the promise and the challenge of generalization across operational regimes.

In contrast to feature engineering approaches, end-to-end methods process complete measurement sequences directly through deep learning architectures. By learning to automatically extract appropriate features from raw voltage, current, and temperature data, these approaches eliminate the need for domain expertise in feature design. These include recurrent neural networks (RNN) that analyze full charge–discharge cycles [17], convolutional neural networks (CNN) that treat voltage-capacity curves as images [18], and hybrid architectures combining CNNs with transformer-based models for improved prediction accuracy [19]. Recent advances have also explored encoder–decoder architectures with CNN or RNN for capacity forecasting, where an encoder network compresses input sequences into compact representations and a decoder network generates predictions from these representations. This approach has shown promise for handling variable-length input sequences and producing multi-step forecasts [20]. Related trends also include attention-based architectures (including transformer variants) that can leverage compact features, albeit sometimes requiring additional sensing such as impedance information [21].

Beyond single-task learning, recent work targets joint objectives and robustness under realistic BMS constraints. Chen et al. jointly predict state of health (SOH) and Remaining Useful Life (RUL) from a single charging cycle using a 2D-CNN feature extractor fused with handcrafted descriptors, reducing errors versus single-task baselines [22]. Robustness to field noise has been improved with anti-noise Long Short-Term Memory (LSTM) variants for RUL estimation [23], while SOC precision has been increased via Bayesian hyperparameter optimization of BiLSTM architectures [24].

However, these data-driven methods face practical limitations that restrict real-world deployment. Differential analysis techniques amplify measurement noise and require high-resolution data with low charge rate, making them unsuitable for battery management systems (BMS) [25]. Methods relying on specific charging segments cannot adapt to variable charging protocols. Full-cycle approaches need complete charge–discharge data, or require signals like SOC that many BMS do not reliably estimate. Furthermore, storing raw time-series data for thousands of cells over a long period of time creates prohibitive memory and bandwidth requirements for BMS [25]. This has motivated data-efficient logging and feature aggregation strategies for fleet-scale analytics, including histogram-based usage-history encodings that are explicitly designed to reduce storage while preserving operational exposure information [2].

Practical constraints necessitate health-monitoring methods that rely solely on measurements already available in the BMS (voltage,

current, and temperature) and minimize storage through compact encoding. Instead of depending solely on carefully hand-crafted features or analyzing individual charge segments, our approach focuses on how operating conditions drive aging: elevated temperature accelerates side reactions, high charge rates increase the risk of lithium plating, and extended periods at high SOC quicken calendar aging [26]. By discretizing the combined voltage, current, and temperature data into bins and accumulating the time spent in each, we compress lengthy time series into a compact exposure summary suitable for fleet-scale capacity forecasting. Importantly, this "usage-history" perspective aligns with recent efforts to bridge laboratory and field data via aggregated representations (e.g., histogram variables) while explicitly managing data-quality limitations [27].

However, most operational-condition approaches do not fully exploit the structure of battery operational data. In particular, many binning-based methods treat voltage, current, and temperature as independent variables and summarize each dimension separately, effectively reducing the joint operating space to a set of one-dimensional marginal statistics [28,29]. As a result, stressor interactions (e.g., high current at low temperature or elevated temperature at high SOC) that govern key degradation mechanisms are not represented, and the resulting features discard temporal information. Yet, degradation is driven not only by cumulative exposure to operating conditions, but also by how frequently and how rapidly the cell transitions between regimes; different transition sequences can activate distinct pathways, even when the total exposure time is similar [30].

This work overcomes these limitations by (i) using joint binning over voltage–current–temperature to preserve cross-variable coupling and (ii) explicitly encoding transitions between operating states to retain temporal dynamics. Building on this transition-aware representation, we develop an encoder–decoder learning framework that maps compact operational histories to long-horizon capacity forecasts.

The key contributions of this work include:

- Development of a transition-aware feature encoding method that addresses two limitations of existing operational condition analysis: the loss of cross-variable coupling when voltage, current, and temperature are binned independently, and the loss of temporal dynamics when only cumulative exposure durations are retained. The proposed encoding captures both joint operational conditions and transition sequences using only standard BMS measurements.
- Development of an encoder–decoder architecture-based neural network that further compresses encoded operational features into learned representations for long-term capacity forecasting.
- Demonstration of 94.3% data storage reduction compared to raw time-series approaches while retaining temporal dynamics, achieving 1.68% MAPE for capacity forecasting.
- Systematic benchmarking against duration aggregation and independent binning approaches, demonstrating that preserving transition dynamics and joint operational conditions reduces worst-case capacity prediction errors from 15.96% to 7.24%, a reduction exceeding 50%.

This paper is organized as follows: Section 2 describes the experimental datasets used for validation. Section 3 presents our joint binning-based feature extraction approach. Section 4 describes the model and algorithms used. Section 5 evaluates the proposed method and compares it against alternative approaches. Finally, Section 6 summarizes the key findings and outlines future research directions.

2. Data

To validate our joint binning-based capacity forecasting approach, we use two datasets that indicate the nonlinear degradation behavior of lithium iron phosphate (LFP)/graphite cells under accelerated aging conditions. Both datasets use identical cell chemistry and experimental protocols, providing a basis for evaluating predictive models across different sample populations.

The Severson dataset contains cycle data[1] for 124 A123 Systems 18650 cells with a nominal capacity of 1.1 Ah [31]. Following an initial formation cycle at C/10,[2] cells were subjected to extremely fast-charging protocols with stepped current profiles ranging from 3C to 8C, charging to 80% SOC. All discharge cycles were performed at 4C, corresponding to complete discharge in 15 min. Data collection spanned three experimental batches using a 48-channel Arbin LBT potentiostat, with thermal chambers maintained at 30 °C to ensure consistent temperature conditions.

The Attia dataset extends this work by providing data for 235 cells, of which 45 were cycled to end-of-life, defined as reaching 80% of nominal capacity [32]. For our analysis, we utilize these 45 cells that underwent degradation trajectories to failure. These cells underwent a standardized 10-minute charging protocol followed by constant voltage phases, maintaining identical environmental and electrochemical conditions to the Severson experiments.

Both datasets show nonlinear capacity fade profiles with knee points (Fig. 1), representing the transition from gradual to rapid degradation under fast-charging conditions. Fig. 1 illustrates the cycle number (n) versus capacity (Q) trajectories for individual cells, indicating significant cell-to-cell variation, reflecting stochastic degradation processes inherent in battery aging. These nonlinear degradation patterns and cell-to-cell variations provide a rigorous test environment for evaluating our joint operational condition analysis approach. The following section describes how we process this data into compact representations that preserve both operational coupling and temporal dynamics.

3. Data processing and feature extraction

3.1. Motivation for joint operational condition encoding

Battery degradation rates vary substantially with operational conditions. High temperatures accelerate chemical reactions, high charge rates may result in lithium plating, and prolonged exposure to high SOC during idle periods accelerates calendar aging [26,33]. Given that degradation accumulates through exposure to these conditions, operational data can be discretized into bins representing distinct usage patterns by partitioning the voltage-current-temperature space. The effectiveness of operational condition binning has been demonstrated in recent battery health forecasting studies. Richardson et al. [28] and Greenbank and Howey [29] showed that time spent in different voltage regions correlates strongly with capacity fade. Zhang et al. [34] developed a systematic histogram-based framework that extracts statistical features from operational data.

However, the aforementioned approaches treat voltage, current, and temperature as independent variables, decomposing the multi-dimensional operational space into separate one-dimensional projections rather than capturing their coupled effects on electrochemical processes. This independent binning approach misses degradation mechanisms emerging from coupled effects. Lithium plating risk, for instance, depends simultaneously on temperature, current magnitude, and voltage conditions that cannot be identified through isolated variable binning [35]. Similarly, calendar aging under high-temperature, zero-current storage conditions shows a distinct degradation trend compared to low-temperature storage [36].

Independent binning also discards temporal dynamics between operational states. Battery degradation depends not only on cumulative

[1] A cycle refers to one complete charge–discharge sequence of a battery cell.

[2] C-rate indicates the charging or discharging current relative to the battery's nominal capacity. For example, 1C corresponds to complete charge or discharge in one hour, while 4C represents complete discharge in 15 min.

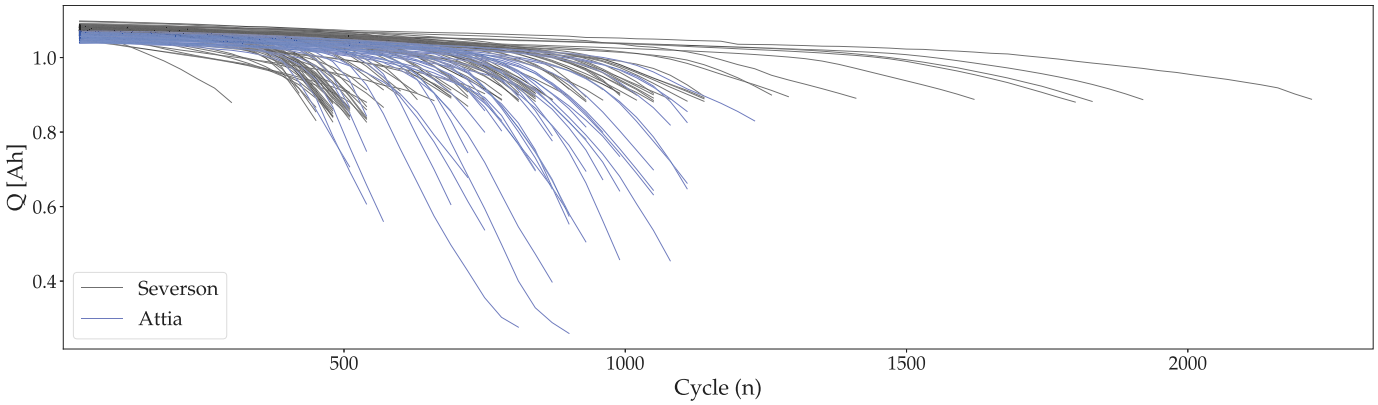


Fig. 1. Cycle number versus capacity for individual cells in the Severson and Attia datasets [31,32].

condition exposure but also on the sequence and rate of transitions between operational regimes. Temperature transitions from cold to hot environments can trigger cascading electrochemical reactions, such as electrolyte gassing catalyzed by prior lithium deposition formed during low-temperature cycling [37]. Different operational sequences can activate distinct degradation pathways even when total exposure to individual conditions remains constant, with path-dependent electrochemical processes influencing aging trajectories [30].

This work addresses both limitations through joint binning that simultaneously considers voltage-current-temperature combinations while explicitly capturing transition dynamics between operational conditions. The following subsections detail the discretization scheme, transition-aware encoding, and comparison with alternative strategies.

3.2. Joint operational space discretization

The encoding process consists of three steps: operational space discretization, bin occupancy calculation, and transition sequence encoding. We first discretize the joint measurement space into distinct operational bins that capture the relationships among voltage, current, and temperature. Then, we encode both the cumulative time spent in each bin and the sequence of transitions between bins.

Each operational parameter is divided into discrete bins:

- Voltage: K_V bins across $[V_{\min}, V_{\max}]$
- Current: K_I bins across $[I_{\min}, I_{\max}]$
- Temperature: K_T bins across $[T_{\min}, T_{\max}]$

This creates $K = K_V \times K_I \times K_T$ joint operational bins. Let $s(t) \in 1, \dots, K$ denote the active joint bin at time t.

3.3. proposed transition-aware bin encoding

For time window ΔT, we construct a transition sequence matrix $\mathbf{U} \in \mathbb{R}^{M \times K}$ that captures both the temporal order of operational bin changes and the duration spent in each bin. Each row $m \in \{1, 2, \dots, M\}$ in matrix $\mathbf{U}$ corresponds to a transition event, where only one element is non-zero:

$$U_{m,k} = \begin{cases} t_k^{(m)}, & \text{if bin } k \text{ was active before transition } m \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

where M is the total number of transitions, and $t_k^{(m)}$ is the duration spent in bin k before the mth transition occurs. The transition matrix $\mathbf{U}$ preserves the chronological order of bin changes while encoding the duration spent in each bin before each transition. This representation captures both the temporal sequence of operational bins and the time spent in each bin.

To illustrate the encoding process, consider a simplified example with $K_V = K_I = K_T = 3$, generating $K = 27$ joint bins. Suppose a battery cycle produces the following operational sequence:

$$\text{Bin } 1 \xrightarrow{t_1=120\text{s}} \text{Bin } 27 \xrightarrow{t_2=85\text{s}} \text{Bin } 1 \xrightarrow{t_3=200\text{s}} \cdots \tag{2}$$

This sequence contains $M = 3$ transitions. The transition matrix $\mathbf{U} \in \mathbb{R}^{3\times 27}$ encodes both the temporal order and duration:

$$\mathbf{U} = \begin{bmatrix} 120 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 85 \\ 200 & 0 & \cdots & 0 \end{bmatrix} \tag{3}$$

Row 1 indicates 120 s in Bin 1 before the first transition; row 2 indicates 85 s in Bin 27 before the second transition; row 3 indicates 200 s in Bin 1 before the third transition.

3.3.1. Bin duration aggregation

From the transition sequence matrix, we can derive an aggregated representation by summing the durations for each bin across all transitions:

$$d_k = \sum_{m=1}^{M} U_{m,k}, \quad \forall k \in 1, \dots, K \tag{4}$$

This bin duration vector $\mathbf{d} \in \mathbb{R}^K$ provides the total time spent in each operational bin, discarding the temporal sequence information while retaining the cumulative exposure profile. For the example sequence in (3), the corresponding bin duration vector $\mathbf{d} \in \mathbb{R}^{27}$ aggregates occupancy times across all visits to each bin: $d_1 = 320\,\text{s}$, $d_{27} = 85\,\text{s}$, with all other entries zero. This aggregation retains the total exposure to each operational condition but discards the transition sequence preserved in $\mathbf{U}$, the information that the cell transitioned from bin 1 to bin 27 and back is lost.

3.3.2. Independent measurement binning

Independent binning discretizes each operational variable separately, constructing marginal histograms:

$$d_k^T = \sum_{t:T(t)\in \text{bin}k} \Delta t, \quad d_j^I = \sum_{t:I(t)\in \text{bin}j} \Delta t, \quad d_i^V = \sum_{t:V(t)\in \text{bin}_i} \Delta t \tag{5}$$

where bin indices i, j, and k correspond to voltage, current, and temperature bins, respectively. This approach generates only $K_V + K_I + K_T$ features.

3.4. comparison with alternative encoding strategies

Fig. 2 compares three feature encoding strategies against time-series data based on computational efficiency and information retention.

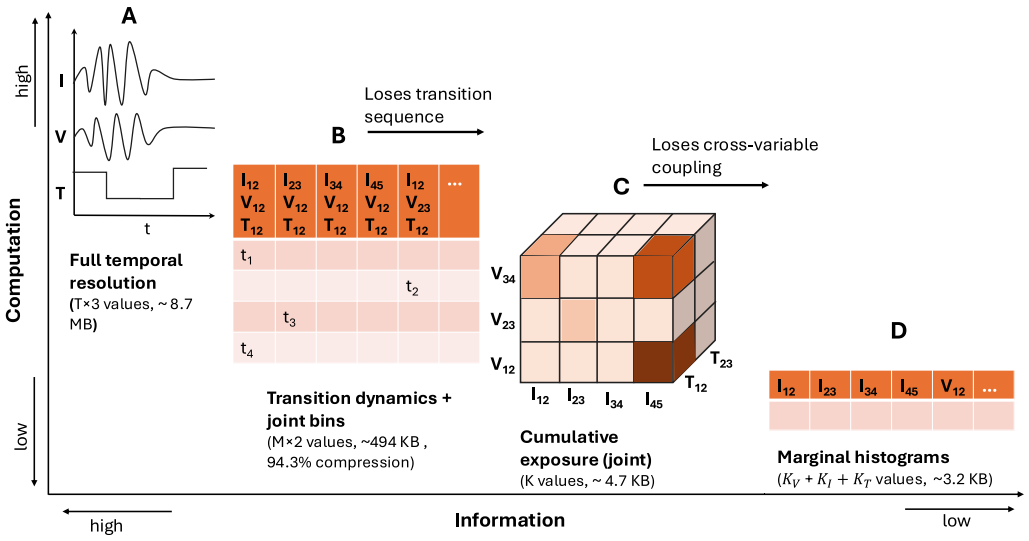


Fig. 2. Comparison of feature encoding strategies illustrating the trade-off between computational burden/storage and information retention. (A) Raw time-series data preserving full temporal resolution ($T \times 3$ values). (B) Proposed transition-aware encoding capturing transition dynamics and joint operational bins ($M \times 2$ values). (C) Bin duration aggregation retaining cumulative exposure in joint bins but discarding transition sequences (K values). (D) Independent measurement binning storing marginal histograms, losing both temporal order and cross-variable dependencies ($K_V + K_I + K_T$ values). The notation X_{ij} denotes the bin spanning thresholds X_i to X_j for variable $X \in \{V$ (voltage), I (current), T (temperature)$\}$.

- **Raw Time-Series (A)**: This approach records every observation, requiring $T \times 3$ storage, where T is the total number of time points and 3 represents the voltage, current, and temperature measurements. While it preserves complete temporal detail, it requires maximum storage.
- **Transition-aware Bin Encoding (B)**: This method stores transition events as pairs of (duration, bin index), requiring $M \times 2$ storage. Since $M \ll T$ typically, this approach significantly reduces storage while retaining temporal dynamics and measurement relationships.
- **Bin Duration Aggregation (C)**: Compresses to cumulative time per joint bin (K values), losing transition sequences.
- **Independent Measurement Binning (D)**: Further reduces to marginal histograms ($K_V + K_I + K_T$ values), additionally losing cross-measurement dependencies.

3.5. summary of proposed encoding method

The proposed transition-aware encoding differs from existing approaches in two key aspects. First, joint binning captures the coupled effects of voltage, current, and temperature by discretizing the three-dimensional operational space into $K = K_V \times K_I \times K_T$ bins, whereas independent binning projects this space onto separate one-dimensional histograms, losing cross-variable dependencies. Second, the transition sequence matrix $\mathbf{U}$ preserves the temporal ordering of operational states, whereas duration aggregation retains only cumulative exposure times.

Given raw measurements $(V(t), I(t), T(t))$ over a cycle, the encoding proceeds as follows:

1. Assign each time point to a joint operational bin $s(t) \in \{1, \ldots, K\}$.
2. Identify transition events where $s(t)$ changes.
3. Construct the transition matrix $\mathbf{U} \in \mathbb{R}^{M \times K}$, where row m records the duration spent in the active bin before the mth transition.

This encoding compresses raw time-series data by a factor proportional to T/M while preserving both operational coupling and temporal dynamics.

4. Model and learning

Having encoded operational data into compact transition-aware representations, we now describe the neural network framework that maps these representations to capacity forecasts. We model battery capacity evolution using an encoder–decoder framework as shown in Fig. 3. This framework builds upon the feature extraction described in Section 3. The model consists of three main components: a feature encoder that compresses the transition matrix representations, an encoder that processes historical data, and a decoder that forecasts capacity.

4.1. Feature encoding and sequence processing

Our encoder–decoder framework processes battery operational data through two stages: (i) compressing high-dimensional transition matrices into compact features using a convolutional network, and (ii) processing these features through a recurrent sequence model that generates capacity predictions. The transition matrix $U_n \in \mathbb{R}^{M_n \times K}$ at cycle n is first compressed using a feature encoder:

$$z_n = \phi(U_n, \theta_\phi), \tag{6}$$

where ϕ is the feature encoder function with parameters θ_ϕ, and $z_n \in \mathbb{R}^d$ is the compressed feature representation with $d \ll M_n \times K$. Note that M_n varies across cycles because different operational patterns during each cycle result in different numbers of transitions between operational bins. For instance, a cycle with frequent switching between charging and discharging will have more transitions (M_n large) than a cycle with steady operation (M_n small). The feature encoder ϕ addresses a key challenge: transition matrices U_n vary in size across cycles (different M_n) and contain high-dimensional sparse representations. The convolutional architecture compresses these variable-length matrices into fixed-dimension feature vectors $z_n \in \mathbb{R}^d$, enabling subsequent sequence processing regardless of the number of transitions in each cycle.

For the historical period ($n = 1, \ldots, n_e - 1$), we work with the sequence $\{(z_n, \Delta Q_n)\}_{n=1}^{n_e-1}$, where z_n are the compressed operational features from (6) and $\Delta Q_n = Q_n - Q_{n-1}$ is the observed capacity change.

To capture history-dependent effects that are not fully specified by instantaneous (V, I, T), we maintain a learned latent state x_n that is intended to summarize slowly evolving internal variables (e.g., loss of

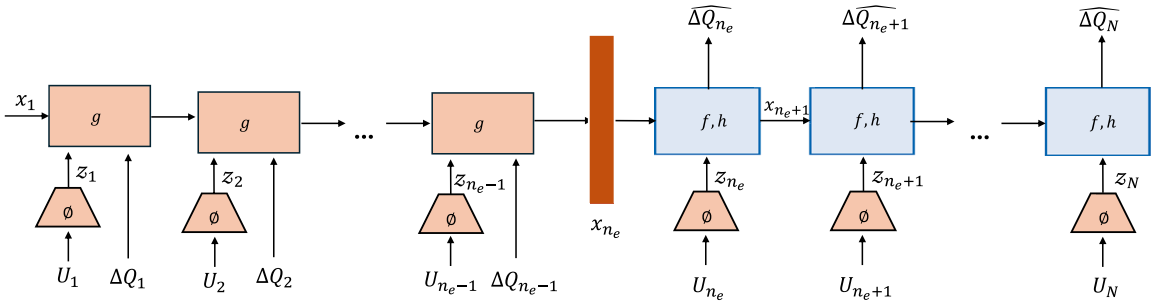


Fig. 3. Encoder–decoder architecture for battery capacity forecasting. The feature encoder ϕ compresses transition matrices U_n into low-dimensional features z_n. The encoder processes these features with capacity changes ΔQ_n through the function g to generate latent states. The decoder uses the final latent state x_{n_e} and future features z_n to recursively predict capacity changes $\widehat{\Delta Q}_n$.

active lithium, impedance growth). We update this state with a history-update function g, which we refer to as the *encoder* because it encodes labeled history (features and measured degradation) into the compact state used for forecasting. During the historical window, g takes the previous state x_n, the features z_n, and the measured ΔQ_n and returns the next state. Using ΔQ_n in the update ties the latent state to the measured degradation, improving identifiability across cells operating under similar usage conditions [38,39].

Formally, the encoder updates the latent state as:

$$x_{n+1} = g(x_n, z_n, \Delta Q_n, \theta_g), \quad n = 1, \dots, n_e - 1, \tag{7}$$

where x_1 is initialized randomly.

For the forecasting period ($n = n_e, \dots, N$), the model predicts future capacity changes using only operational features, as true capacity measurements are unavailable. The decoder functions f and h update the latent state and generate capacity predictions:

$$\begin{aligned} x_{n+1} &= f(x_n, z_n, \theta_f), \\ \widehat{\Delta Q}_n &= h(x_n, z_n, \theta_h), \quad n = n_e, \dots, N. \end{aligned} \tag{8}$$

where $\widehat{\Delta Q}_n$ denotes the predicted capacity change at cycle n.

The complete prediction pipeline can be summarized mathematically as:

$$(V(t), I(t), T(t)) \xrightarrow{\text{Encoding}} \mathbf{U}_n \xrightarrow{\phi} z_n \xrightarrow{g/f} x_n \xrightarrow{h} \widehat{\Delta Q}_n \tag{9}$$

where each arrow represents a transformation: raw measurements are encoded into the transition matrix $\mathbf{U}_n$, compressed by the feature encoder ϕ into features z_n, processed by the encoder g (historical period) or decoder f (forecasting period) to update the latent state x_n, and finally mapped by h to the predicted capacity change $\widehat{\Delta Q}_n$.

The following section describes the type of neural networks used to implement the functions ϕ, g, f, and h.

4.2. Neural network architecture

4.2.1. Feature extraction network

The feature encoder ϕ (Eq. (6)) is implemented as a deep convolutional neural network as shown in Fig. 4. Each transition matrix U_n is processed through sequential blocks containing:

- 1D Convolution: Applies learnable kernels to extract local temporal patterns from the input sequences [40].
- ReLU Activation: Introduces non-linearity via the rectified linear unit function $\max(0, x)$ [41].
- Dropout: Randomly deactivates neurons with probability p during training to prevent overfitting [42].

The 1D convolutional layers process the temporal structure of battery operational transitions through sliding kernels. As shown in Fig. 4, the stacked convolutional blocks form a hierarchical structure

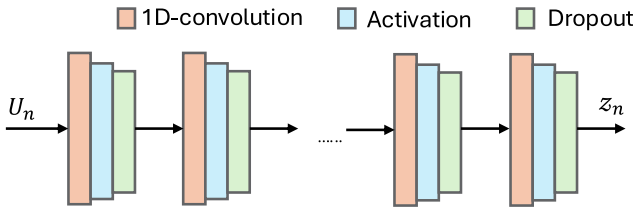


Fig. 4. Feature encoder (ϕ) architecture. The network processes transition matrix U_n through sequential Convolution-ReLU-Dropout blocks to extract compressed feature representations z_n.

where deeper layers capture increasingly abstract features [43]. This architecture progressively reduces the dimensionality from the high-dimensional input $U_n \in \mathbb{R}^{M_n \times K}$ to the compact representation $z_n \in \mathbb{R}^d$ where $d \ll M_n \times K$, with the network learning which features to extract through end-to-end training [44].

4.2.2. Sequence processing networks

Functions g, f, and h are implemented using a Long Short-Term Memory (LSTM) network. LSTMs use gating mechanisms, specifically input, forget, and output gates, to selectively retain or discard information [45]. This gating architecture enables the modeling of long-range temporal dependencies, which is particularly important for battery degradation processes that span hundreds of cycles [46].

The encoder function g (Eq. (7)) is realized as an LSTM that processes the historical data. At each timestep $n \in \{1, \dots, n_e - 1\}$ it takes the current state x_n, compressed features z_n from Eq. (6), and observed capacity change ΔQ_n as inputs to produce the next state x_{n+1}.

The decoder function f (Eq. (8)) is implemented as an LSTM that updates states based solely on features z_n, since actual capacity measurements are unavailable during prediction. The function h in Eq. (8) is a fully connected layer that combines the current state x_n and features z_n to output the predicted capacity change $\widehat{\Delta Q}_n$.

4.3. Learning

The complete parameter vector $\theta = [\theta_\phi^\top, \theta_g^\top, \theta_f^\top, \theta_h^\top]^\top$ is optimized by minimizing the mean absolute error (MAE) between predicted capacity changes $\widehat{\Delta Q}_n$ (Eq. (8)) and observed values ΔQ_n for $n \in \{n_e, \dots, N\}$, chosen for its robustness to outliers in capacity measurements:

$$\mathcal{L}(\theta) = \frac{1}{N - n_e + 1} \sum_{n=n_e}^{N} \left| \Delta Q_n - \widehat{\Delta Q}_n \right|. \tag{10}$$

The model is trained end-to-end using backpropagation through time (BPTT) [47], which computes gradients across the entire encoder–decoder sequence. Parameters are updated using the Adam optimizer [48]. Early stopping based on validation loss is applied to prevent overfitting [49].

Table 1
Discretization scheme for voltage, current, and temperature. Intervals use standard notation where $[a, b)$ denotes $a \leq x < b$.

Quantity	Bin label	Range
Voltage	V_{12}	[1.9, 3.0] V
	V_{23}	(3.0, 3.6] V
	V_{34}	(3.6, 4.0] V
Current	I_{12}	[−4.0, −0.01] A
	I_{23}	(−0.01, 0.01] A
	I_{34}	(0.01, 2] A
	I_{45}	(2, 8] A
Temperature	T_{12}	[20, 40] °C

5. Experimental setup and results

This section describes the experimental setup, performance evaluation, and comparative analysis of the three feature-encoding strategies for battery capacity forecasting. The comparison quantifies the benefits of preserving cross-variable coupling and temporal dynamics in operational condition encoding.

5.1. Experimental setup

The following outlines the discretization scheme, data preparation procedures, and evaluation framework used across all experiments.

5.1.1. Bin configuration

The discretization scheme for voltage, current, and temperature is presented in Table 1. These bin boundaries were determined through analysis of the empirical distributions in the dataset combined with hyperparameter tuning to balance feature resolution against computational complexity.

The voltage range is divided into three bins spanning the typical operational voltage range of LFP cells. The current discretization uses four bins to distinguish between different charge/discharge rates, with finer resolution near zero to capture rest periods and calendar aging effects. Temperature uses a single bin ($K_T = 1$) spanning the observed range $[20, 40]$°C. Alternative configurations with $K_T \in \{2, 3\}$ were evaluated but yielded no improvement. Since the temperature data indicates a narrow distribution, increasing K_T expands the total bin count $K = K_V K_I K_T$ from 12 to 24 or 36 without improving forecast performance for this dataset. Consequently, we maintain $K_T = 1$, though datasets with broader or multimodal temperature distributions may benefit from higher values.

The selected configuration ($K_V = 3$, $K_I = 4$, $K_T = 1$) generates $K = 12$ operational bins. This configuration balances sufficient resolution to distinguish degradation-relevant operational regimes against the need for compact representations. This discretization granularity directly influences transition matrix complexity: finer partitioning increases the number of possible state transitions, leading to larger M_n values and correspondingly higher-dimensional feature representations. This design choice reflects the fundamental trade-off between capturing detailed operational dynamics and maintaining computational efficiency.

5.1.2. Window generation and padding

For each cell with total cycle count N, we create encoder–decoder pairs at three historical context ratios $r \in \{0.30, 0.50, 0.80\}$, where r represents the fraction of the cell's lifetime used as historical context for the encoder. The encoding boundary is set at

$$n_e = \lfloor rN \rfloor,$$

with the forecast horizon spanning the remaining $n_d = N - n_e$ cycles. This creates three forecasting scenarios per cell: long-horizon (30:70), medium-horizon (50:50), and short-horizon (80:20) context-to-prediction ratios.

Since cells show varying numbers of transitions M_n at each cycle, transition matrices must be standardized to a uniform size for batch processing in neural networks [44]. Padding refers to the addition of placeholder values to create consistent matrix sizes across all samples in a training batch. We pad all matrices to M_{max} rows (where M_{max} is the maximum number of transitions observed across the dataset) by appending zero-valued rows, for consistent matrix dimensions.

Similarly, encoder and decoder sequences vary in length across cells due to different values of n_e and n_d. To enable batch processing, sequences are padded with zero tokens to uniform lengths. Following established practices for sequence-to-sequence architectures, which map input sequences to output sequences of potentially different lengths [50], we apply left-padding to encoder sequences (adding zeros at the beginning) to preserve the most recent operational history near the encoding boundary. Decoder sequences receive right-padding (adding zeros at the end) to maintain the chronological order of predictions. The padding elements, referred to as zero tokens, are placeholder values that do not represent actual operational data. During training, these padding tokens are masked – excluded from gradient computation – to prevent them from influencing the model's parameter updates [51].

5.1.3. Data split

After generating the encoder–decoder pairs described above, the cells are randomly divided *cell-wise* into 40% for testing and 60% for training/validation. The training/validation portion is further randomly split into 70% for training and 30% for validation. This cell-wise partitioning ensures that all windows originating from a given cell stay in the same subset, preventing information leakage across splits.

5.1.4. Evaluation criteria

The model's performance is assessed using the following performance metrics: root mean squared error (RMSE) and mean absolute percentage error (MAPE):

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}, \tag{11}$$

$$MAPE = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{y_i - \hat{y}_i}{y_i}\right|, \tag{12}$$

where N represents the number of samples, y_i denotes the true value, and $\hat{y}_i$ the predicted value. For capacity change predictions (ΔQ), MAPE uses the absolute capacity Q_i as the denominator rather than ΔQ_i to avoid numerical instability when capacity changes approach zero:

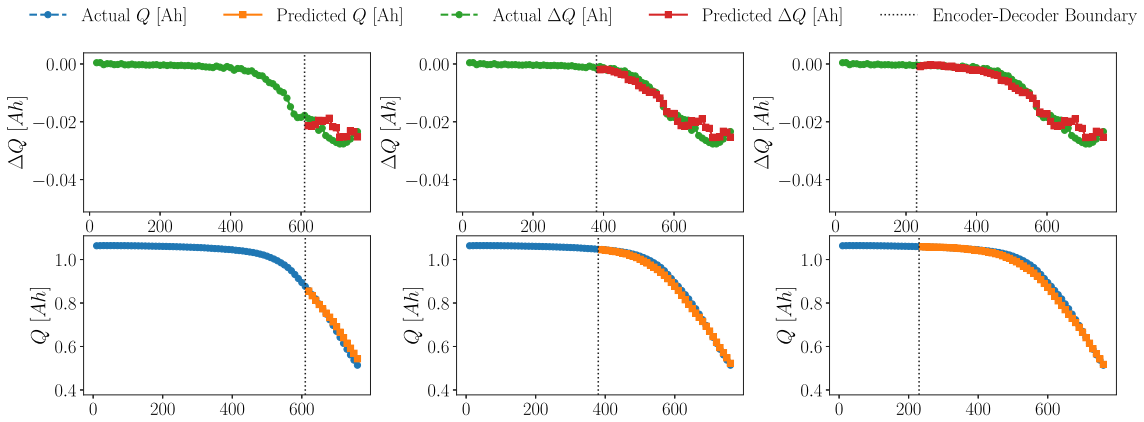
$$\text{MAPE}_{\Delta Q} = \frac{100}{N}\sum_{i=1}^{N}\left|\frac{\Delta Q_i - \widehat{\Delta Q}_i}{Q_i}\right|. \tag{13}$$

5.2. Results

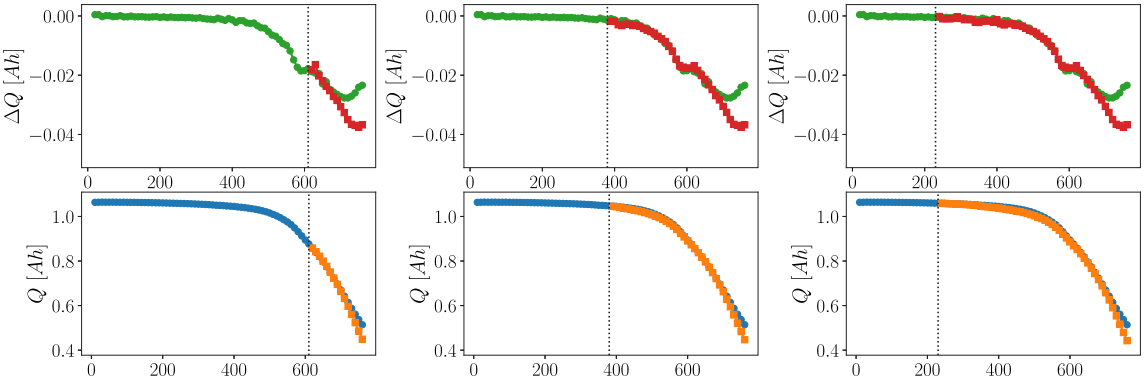
The following sections evaluate the proposed transition-aware encoding against duration aggregation and independent binning approaches in terms of forecasting accuracy, error distributions, and data storage requirements.

5.2.1. Capacity-forecasting

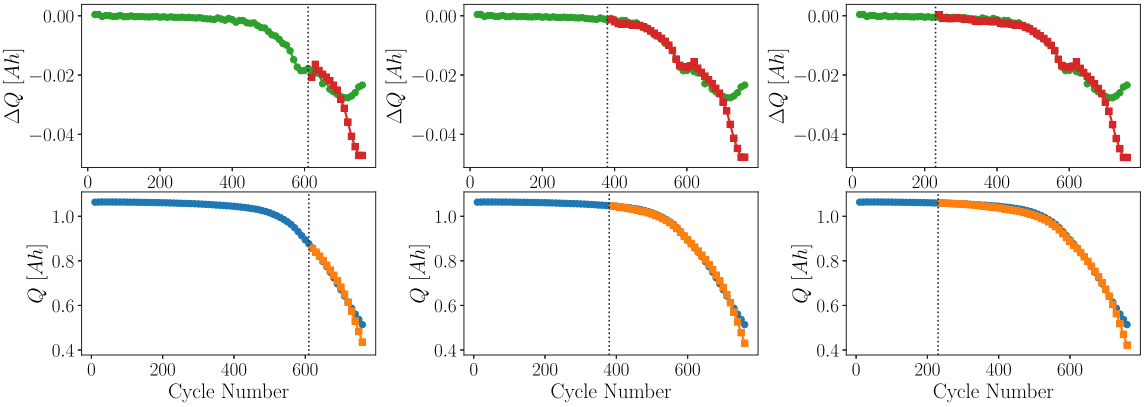
The encoder–decoder models were evaluated on their ability to predict battery capacity evolution across different forecasting horizons. Predictions were generated at 10-cycle intervals rather than every cycle to mitigate error propagation in long-term forecasts. Single-cycle capacity changes (ΔQ) are typically small and susceptible to measurement noise. The cumulative summation of individual prediction errors can lead to significant degradation in forecast accuracy over extended horizons. Additionally, predicting at every cycle increases computational overhead by extending encoder–decoder sequence lengths. This



(a) Transition-aware encoding: Forecasts of capacity difference (ΔQ, top) and total capacity (Q, bottom), with forecasting horizons increasing from left to right.



(b) Duration aggregation: Forecasts of capacity difference (ΔQ, top) and total capacity (Q, bottom), with forecasting horizons increasing from left to right.



(c) Independent measurement binning: Forecasts of capacity difference (ΔQ, top) and total capacity (Q, bottom), with forecasting horizons increasing from left to right.

Fig. 5. Comparison of capacity forecasts across three feature encoding strategies: (a) transition-aware encoding, (b) duration aggregation, and (c) independent measurement binning. Each shows capacity difference (ΔQ, top) and total capacity (Q, bottom) predictions with increasing forecast horizons from left to right.

10-cycle interval represents a tunable hyperparameter that can be optimized based on dataset characteristics and forecasting requirements. Under this approach, capacity at cycle n is computed as:

$$Q_n = Q_{n_e} + \sum_{i=n_e+10}^{n} \widehat{\Delta Q}_i, \quad n \in \{n_e + 10, n_e + 20, \ldots, N\} \tag{14}$$

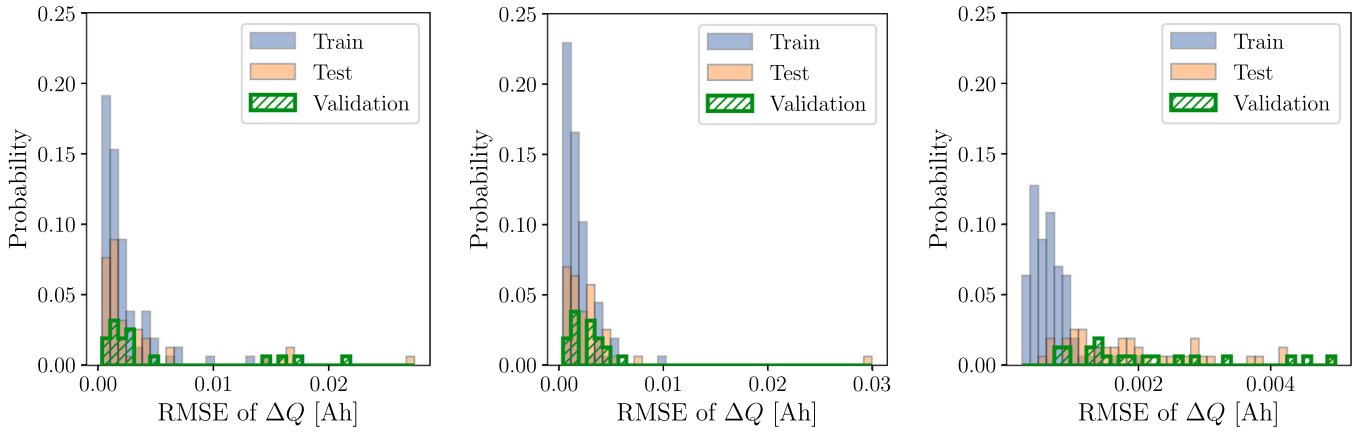
where Q_{n_e} represents the known capacity at the encoding boundary and $\widehat{\Delta Q}_i$ denotes the predicted capacity change at cycle i.

Fig. 5 compares predicted and measured capacity trajectories for a representative test cell at three forecast windows that span 20%, 50%, and 70% of the cell lifetime. All models output the incremental capacity change $\widehat{\Delta Q}$ and reconstruct the absolute capacity Q via (14). The cell shows a clear knee at roughly 600 cycles, a transition point where capacity degradation shifts from gradual to rapid decline, followed by a strongly *non-linear tail* in which ΔQ accelerates [52].

Transition-aware encoding as shown in Fig. 5(a) tracks both the gradual pre-knee fade and the accelerated ΔQ tail with high accuracy across all three windows. By contrast, duration aggregation (Fig. 5(b)) and independent binning(Fig. 5(c)) begin to diverge once ΔQ departs

Table 2
Comparison of capacity forecasting performance across three feature encoding strategies on the test set: transition-aware encoding, duration aggregation, and independent binning on the test set. Overall, worst-case and best-case metrics are reported for both absolute capacity (Q) and capacity change (ΔQ).

Feature Encoding	Overall				Worst				Best			
	[b]MAPE (%)		RMSE		[b]MAPE (%)		RMSE		[b]MAPE (%)		RMSE	
	Q	ΔQ	Q	ΔQ	Q	ΔQ	Q	ΔQ	Q	ΔQ	Q	ΔQ
Transition-aware encoding	1.68	0.16	0.0250	0.0023	7.24	0.44	0.0850	0.0050	0.25	0.04	0.0031	0.0005
Duration aggregation	2.32	0.20	0.0386	0.0066	12.24	0.94	0.1278	0.0299	0.21	0.04	0.0025	0.0005
Independent binning	2.48	0.21	0.0471	0.0067	15.96	1.07	0.1925	0.0274	0.22	0.04	0.0027	0.0005



(a) Independent binning (b) Duration aggregation (c) Transition-aware encoding

Fig. 6. Probability distributions of the RMSE in forecasting the capacity change (ΔQ) for the training, validation, and test sets, comparing the three feature-encoding strategies: independent binning, duration aggregation, and the proposed transition-aware encoding.

from its initial linear trend, ultimately underestimating the remaining capacity at end-of-life.

The performance difference between encoding strategies can be attributed to their information retention characteristics. Transition-aware encoding preserves the sequence of operational states, enabling the model to identify degradation patterns associated with specific transition sequences (e.g., rapid charging following low-temperature operation). Duration aggregation retains joint bin occupancy but cannot distinguish between different orderings of the same operational conditions. Independent binning further loses the coupling between variables, preventing identification of degradation mechanisms that depend on simultaneous voltage-current-temperature conditions.

Table 2 presents quantitative performance metrics across the test set. The transition-aware encoding achieves the lowest overall errors with MAPE values of 1.68% for absolute capacity and 0.16% for capacity changes. Notably, the worst-case performance shows even larger differences between methods: transition-aware encoding limits worst-case MAPE to 7.24% for Q, while independent binning reaches 15.96%. These results confirm that encoding temporal dynamics and cross-signal dependencies improves forecast accuracy.

Fig. 6 further illustrates the RMSE error distributions in forecasting ΔQ across all cells in the training, validation, and test sets. The Fig. 6(c) shows that the transition-aware encoding has a highly concentrated error distribution with most predictions achieving RMSE below 0.005 Ah. In contrast, duration aggregation (Fig. 6(b)) and independent binning (Fig. 6(a)) show wider distributions with tails extending beyond 0.020 Ah, indicating less reliable predictions across the cell population.

5.2.2. Data compression and storage efficiency

The transition-aware encoding represents each cell's operational history as a sequence of M transitions, stored as pairs (k_m, t_m) where k_m is the bin index and t_m is the time spent in that bin. This corresponds to the transition matrix $\mathbf{U} \in \mathbb{R}^{M \times K}$ described in Section 3.3, which can be stored compactly as $2M$ values.

Table 3
Data storage comparison for a random cell with 1008 cycles.

Representation	Values stored	Size (float32)
Raw time series (V, I, T)	2,217,672	8.7 MB
Transition-aware encoding	126,414	494 KB
Duration aggregation (joint)	1212	4.7 KB
Independent binning	808	3.2 KB

Table 3 compares storage requirements across all four data representations for a representative cell from the dataset. Raw time-series storage requires three floating-point values (voltage, current, temperature) at each timestep, resulting in $3T$ total values. The transition-aware encoding achieves 94.3% compression relative to raw data while preserving the sequence of operational transitions. Duration aggregation and independent binning group data into 10-cycle windows, storing one histogram per window. Duration aggregation stores the cumulative time in each joint bin per window (101 windows ×12 bins = 1212 values), achieving 99.95% compression but discarding all temporal ordering. Independent binning stores separate histograms for each measurement per window (101 windows ×eight marginal bins = 808 values), achieving 99.96% compression but losing the coupling between voltage, current, and temperature.

For the example cell with $T = 739,224$ samples collected over 1008 cycles, $M = 63{,}207$ transitions occurred between operational bins, producing a ratio $M/T = 0.085$. This demonstrates that operational patterns involve relatively few transitions compared to the total number of measurements, making transition-aware encoding particularly effective for long-term battery monitoring applications where both data compression and preservation of temporal dynamics are essential.

6. Conclusion

This work addresses two fundamental limitations of existing operational condition analysis for battery capacity forecasting: the loss

of cross-variable coupling when voltage, current, and temperature are binned independently, and the loss of temporal dynamics when only cumulative exposure durations are retained. The proposed transition-aware encoding method resolves both limitations by discretizing the joint voltage-current-temperature space into operational bins and tracking the sequence of transitions between bins alongside the duration spent in each. This representation preserves the coupled effects of operational variables and the temporal ordering of condition changes, both of which influence degradation pathways.

An encoder–decoder neural network processes these compact representations to generate long-term capacity forecasts. On lithium–iron–phosphate cells showing nonlinear degradation patterns, the method achieves mean absolute percentage errors of 1.68% for capacity and 0.16% for cycle-to-cycle capacity variation while reducing storage by 94.3% compared to raw time series. Systematic comparison against duration aggregation and independent binning demonstrates that preserving transition dynamics and cross-variable dependencies reduces worst-case capacity prediction errors for a battery from 15.96% to 7.24%, a reduction exceeding 50%. Using only standard battery measurements, voltage, current, and temperature, the method provides a practical solution for cloud-based battery management systems where both storage efficiency and prediction accuracy are essential. Future work will extend the framework to different battery chemistries, incorporate prediction confidence intervals for risk-aware battery management decisions, and investigate adaptive bin configurations that adjust to varying operational patterns.

CRediT authorship contribution statement

Tushar Desai: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Riccardo M.G. Ferrari:** Writing – review & editing, Validation, Supervision, Resources, Funding acquisition, Formal analysis, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This document contains the results of the research project funded by Volvo AB (Sweden), and NWO (Netherlands) under the TKI-HTSM grant 21.0056.

Data availability

Data will be made available on request.

References

[1] Manthiram A. An outlook on lithium ion battery technology. ACS Cent Sci 2017;3(10):1063–9.
[2] Westman K, Aitio A, Nilsson V, Johansson P. The battery information format (BIF): A memory-efficient standard for communicating cell usage and status for battery digital twins (BDTs). J Power Sources 2025;652:237538. http://dx.doi.org/10.1016/j.jpowsour.2025.237538.
[3] Edge JS, O'Kane S, Prosser R, Kirkaldy ND, Patel AN, Hales A, et al. Lithium ion battery degradation: what you need to know. Phys Chem Chem Phys 2021;23(14):8200–21.
[4] Kim K, Choi J-I. Effect of cell-to-cell variation and module configuration on the performance of lithium-ion battery systems. Appl Energy 2023;352(121888):121888.
[5] Yang F, Xie Y, Deng Y, Yuan C. Predictive modeling of battery degradation and greenhouse gas emissions from U.S. state-level electric vehicle operation. Nat Commun 2018;9(1):2429.
[6] Lanubile A, Bosoni P, Pozzato G, Allam A, Acquarone M, Onori S. Domain knowledge-guided machine learning framework for state of health estimation in lithium-ion batteries. Commun Eng 2024;3:168. http://dx.doi.org/10.1038/s44172-024-00304-2.
[7] Wood KN, Kazyak E, Chadwick AF, Chen K-H, Zhang J-G, Thornton K, et al. Dendrites and pits: Untangling the complex behavior of lithium metal anodes through operando video microscopy. ACS Cent Sci 2016;2(11):790–801.
[8] Miguel E, Plett GL, Trimboli MS, Oca L, Iraola U, Bekaert E. Review of computational parameter estimation methods for electrochemical models. J Energy Storage 2021;44:103388.
[9] Lai X, Wang S, Ma S, Xie J, Zheng Y. Parameter sensitivity analysis and simplification of equivalent circuit model for the state of charge of lithium-ion batteries. Electrochim Acta 2020;330:135239.
[10] Wang S, Gao H, Takyi-Aninakwa P, Guerrero JM, Fernandez C, Huang Q. Improved multiple feature-electrochemical thermal coupling modeling of lithium-ion batteries at low-temperature with real-time coefficient correction. Prot Control Mod Power Syst 2024;9(3):157–73.
[11] Campestrini C, Kosch S, Jossen A. Influence of change in open circuit voltage on the state of charge estimation with an extended Kalman filter. J Energy Storage 2017;12:149–56.
[12] Pozzato G, Allam A, Onori S. Lithium-ion battery aging dataset based on electric vehicle real-driving profiles. Data Brief 2022;41:107995.
[13] Zhu J, Dewi Darma MS, Knapp M, Sørensen DR, Heere M, Fang Q, et al. Investigation of lithium-ion battery degradation mechanisms by combining differential voltage analysis and alternating current impedance. J Power Sources 2020;448(227575):227575.
[14] Feng X, Weng C, He X, Han X, Lu L, Ren D, et al. Online state-of-health estimation for Li-ion battery using partial charging segment based on support vector machine. IEEE Trans Veh Technol 2019;68(9):8583–92.
[15] Li W, Zhang H, van Vlijmen B, Dechent P, Sauer DU. Forecasting battery capacity and power degradation with multi-task learning. Energy Storage Mater 2022;53:453–66. http://dx.doi.org/10.1016/j.ensm.2022.09.013.
[16] Li T, Zhou Z, Thelen A, Howey DA, Hu C. Predicting battery lifetime under varying usage conditions from early aging data. Cell Rep Phys Sci 2024;5:101891. http://dx.doi.org/10.1016/j.xcrp.2024.101891.
[17] Lu J, Xiong R, Tian J, Wang C, Hsu C-W, Tsou N-T, et al. Battery degradation prediction against uncertain future conditions with recurrent neural network enabled deep learning. Energy Storage Mater 2022;50:139–51.
[18] Costa N, Sánchez L, Anseán D, Dubarry M. Li-ion battery degradation modes diagnosis via Convolutional Neural Networks. J Energy Storage 2022;55(105558):105558.
[19] Lianpo L, Songmei D, Lin W. Capacity degradation prediction of electric vehicle battery by integrating convolutional neural network with informer model. J Power Sources 2025;651(237497):237497.
[20] Gong Q, Wang P, Cheng Z. An encoder-decoder model based on deep learning for state of health estimation of lithium-ion battery. J Energy Storage 2022;46(103804):103804.
[21] Liu R, Zhang D, Wang L, Mi CC, Bauer P, Qin Z. A transformer-enhanced framework for lithium-ion battery capacity estimation using limited imaginary impedance feature. J Energy Storage 2025;116313. http://dx.doi.org/10.1016/j.est.2025.116313.
[22] Chen J, Li P, Wu L. Joint prediction of SOH and RUL of lithium-ion batteries using single-cycle charging data. Energy (Oxf) 2025;336(138351):138351.
[23] Wang S, Fan Y, Jin S, Takyi-Aninakwa P, Fernandez C. Improved anti-noise adaptive long short-term memory neural network modeling for the robust remaining useful life prediction of lithium-ion batteries. Reliab Eng Syst Saf 2023;230(108920):108920.
[24] Wang S, Ma C, Gao H, Deng D, Fernandez C, Blaabjerg F. Improved hyperparameter Bayesian optimization-bidirectional long short-term memory optimization for high-precision battery state of charge estimation. Energy (Oxf) 2025;328(136598):136598.
[25] Sulzer V, Mohtat P, Aitio A, Lee S, Yeh YT, Steinbacher F, et al. The challenge and opportunity of battery lifetime prediction from field data. Joule 2021;5(8):1934–55.
[26] Leijon J. Charging strategies and battery ageing for electric vehicles: A review. Energy Strat Rev 2025;57(101641):101641.
[27] Steininger V, Rumpf K, Hüsson P, Li W, Sauer DU. Automated feature extraction to integrate field and laboratory data for aging diagnosis of automotive lithium-ion batteries. Cell Rep Phys Sci 2023;4(10):101596. http://dx.doi.org/10.1016/j.xcrp.2023.101596.
[28] Richardson RR, Osborne MA, Howey DA. Battery health prediction under generalized conditions using a Gaussian process transition model. J Energy Storage 2019;23:320–8.
[29] Greenbank S, Howey D. Automated feature extraction and selection for data-driven models of rapid battery capacity fade and end of life. IEEE Trans Ind Inf 2022;18(5):2965–73.
[30] Kim I, Kim H, An S, Oh J, Kim M, Choi JW. Degradation path prediction of lithium-ion batteries under dynamic operating sequences. Energy Env Sci 2025;18(8):3784–94.

[31] Severson KA, Attia PM, Jin N, Perkins N, Jiang B, Yang Z, et al. Data-driven prediction of battery cycle life before capacity degradation. Nature Energy 2019;4(5):383–91.
[32] Attia PM, Grover A, Jin N, Severson KA, Markov TM, Liao Y-H, et al. Closed-loop optimization of fast-charging protocols for batteries with machine learning. Nature 2020;578(7795):397–402.
[33] Liu W, Zheng J, Zhang Z, Gu J, Chen Z, Jiang H, et al. The capacity decay mechanism of the 100% SOC LiCoO2/graphite battery after high-temperature storage. J Power Sources 2023;580(233330):233330.
[34] Zhang Y, Wik T, Bergström J, Pecht M, Zou C. A machine learning-based framework for online prediction of battery ageing trajectory and lifetime using histogram data. J Power Sources 2022;526(231110):231110.
[35] Lin X, Khosravinia K, Hu X, Li J, Lu W. Lithium plating mechanism, detection, and mitigation in lithium-ion batteries. Prog Energy Combust Sci 2021;87(100953):100953.
[36] Dubarry M, Qin N, Brooker P. Calendar aging of commercial Li-ion cells of different chemistries – A review. Curr Opin Electrochem 2018;9:106–13.
[37] Chang W, Bommier C, Fair T, Yeung J, Patil S, Steingart D. Understanding adverse effects of temperature shifts on Li-ion batteries: An operando acoustic study. J Electrochem Soc 2020;167(9):090503.
[38] Galatro D, Romero DA, Freitez JA, Da Silva C, Trescases O, Amon CH. Modeling degradation of lithium-ion batteries considering cell-to-cell variations. J Energy Storage 2021;44(103478):103478.
[39] Plett GL. Extended Kalman filtering for battery management systems of LiPB-based HEV battery packs: Part 3. State and parameter estimation. J Power Sources 2004;134(2):277–92.
[40] Lecun Y, Bottou L, Bengio Y, Haffner P. Gradient-based learning applied to document recognition. Proc IEEE Inst Electr Electron Eng 1998;86(11):2278–324.
[41] Glorot X, Bordes A, Bengio Y. Deep sparse rectifier neural networks. 15, 2011, p. 315–23, AISTATS,
[42] Srivastava N, Hinton GE, Krizhevsky A, Sutskever I, Salakhutdinov R. Dropout: a simple way to prevent neural networks from overfitting. J Mach Learn Res 2014;15(56):1929–58.
[43] Szegedy C, Liu W, Jia Y, Sermanet P, Reed S, Anguelov D, et al. Going deeper with convolutions. In: 2015 IEEE conference on computer vision and pattern recognition. IEEE; 2015, p. 1–9.
[44] LeCun Y, Bengio Y, Hinton G. Deep learning. Nature 2015;521(7553):436–44.
[45] Hochreiter S, Schmidhuber J. Long short-term memory. Neural Comput 1997;9(8):1735–80.
[46] Zhao S, Zhang C, Wang Y. Lithium-ion battery capacity and remaining useful life prediction using board learning system and long short-term memory neural network. J Energy Storage 2022;52(104901):104901.
[47] Werbos PJ. Backpropagation through time: what it does and how to do it. Proc IEEE 1990;78(10):1550–60.
[48] Kingma DP, Ba J. Adam: A method for stochastic optimization. In: International Conference for Learning Representations. 2014.
[49] Prechelt L. Automatic early stopping using cross validation: quantifying the criteria. Neural Netw 1998;11(4):761–7.
[50] Sutskever I, Vinyals O, Le QV. Sequence to sequence learning with neural networks. In: Proceedings of the 27th international conference on neural information processing systems, vol. 2, Cambridge, MA, USA: MIT Press; 2014, p. 3104–12.
[51] Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, et al. Attention is all you need. 2017, ArXiv [Cs.CL].
[52] Fermín-Cueto P, McTurk E, Allerhand M, Medina-Lopez E, Anjos MF, Sylvester J, et al. Identification and machine learning prediction of knee-point and knee-onset in capacity degradation curves of lithium-ion cells. Energy AI 2020;1(100006):100006.