Bayesian CNN-LSTM for Battery State of Health (SoH) Estimation under Randomized Usage Conditions

Saurabh Jaywant Desai[1], Muralikrishnan Ramanujam[2], and Venkataramana Runkana[3]

[1, 2, 3] *TCS Research, Pune, Maharashtra, 411028, India*
sj.desai@tcs.com
r.muralikrishnan2@tcs.com
venkat.runkana@tcs.com

Abstract

The accurate assessment of State of Health (SoH) of battery plays pivotal role in enabling the safe and dependable operation of lithium-ion battery systems. In this work, we propose a hybrid deep learning architecture integrating Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks within a Bayesian learning paradigm to improve SoH prediction accuracy considering complex charge-discharge patterns. The CNN module automatically extracts temporal and local degradation features from sequential charge-discharge profiles, while the LSTM component captures long-term dependencies in the cycling behavior. Bayesian inference is incorporated to quantify predictive uncertainty; posterior sampling yields 90% prediction intervals ($S = 100$; $\mu \pm 1.64\sigma$), which support uncertainty-informed assessment under irregular and noisy inputs. The model is trained and evaluated on the NASA randomized battery usage dataset; labels are computed from capacities at consecutive reference discharge cycles as the incremental change in SoH ($dSoH$), with absolute SoH reconstructed by accumulation at inference. Experimental results demonstrate that the proposed Bayesian CNN-LSTM architecture achieves high prediction accuracy, with an R^2 score of 0.932 and an RMSE of 0.022, while providing uncertainty estimates with reasonable empirical coverage under diverse operational conditions. These findings indicate potential applicability in battery management workflows.

Key Words: Convolutional Neural Networks (CNN), Long Short-Term Memory (LSTM) networks, Bayesian, State of Heath (SoH).

* Corresponding author: Venkataramana Runkana (venkat.runkana@tcs.com)
https://doi.org/10.36001/IJPHM.2026.v17i2.4743

1. Introduction

Conventional internal combustion engine-based vehicles are responsible for approximately 10% of greenhouse gas emissions (Reitz et al., 2019). Governments worldwide are trying to control emissions by coming up with stringent norms or by promoting alternative energy sources. Lithium-ion batteries for energy storage along with renewable energy sources like solar and wind are proving to be prominent alternatives to conventional fossil fuel-based energy sources owing to their long lifecycle and high energy density. However, battery performance degrades over time, leading to capacity fade and reduced efficiency due to increase in internal resistance. The State of Health (SoH) of a battery, defined as the ratio of its current capacity to its rated capacity, is a crucial metric that is a reflection of cycling patterns, the remaining useful life of the battery and indicates the necessity for maintenance or replacement. When a battery deteriorates, its internal resistance increases and nominal capacity decreases. Deterioration in battery SoH is a complex phenomenon. Change in health occurs largely due to the degradation of electrochemical components of the battery. It is a direct function of several parameters which include battery chemistry, charging and discharging patterns, and operating conditions like temperature.

Accurate SoH estimation is essential for avoiding failures and for providing feedback that can help in extending battery life. Traditional approaches for SoH estimation include electrochemical modeling (Ramadass, Haran, White, & Popov, 2003; Bole, Kulkarni, & Daigle, 2014a) and empirical curve-fitting methods (Dubarry, Svoboda, Hwu, & Liaw, 2008; Wei, Dong, & Chen, 2018). While effective under controlled conditions, these methods require detailed battery parameters, are computationally intensive, and they often assume fixed cycling profiles, making them less applicable for dynamic, real-world environments.

Data-driven techniques have gained traction in recent years due to their adaptability and predictive capabilities. Machine learning methods, such as support vector regression

(Nagulapati et al., 2021), Gaussian process regression (Richardson, Osborne, & Howey, 2019, 2017; Dong, Chen, Wei, & Ling, 2018), Brownian motion with particle filter (Dong et al., 2018), and relevance vector regression (Saha, Goebel, Poll, & Christophersen, 2009; Wang, Miao, & Pecht, 2013), have shown promising results. More recently, deep learning approaches have demonstrated superior performance, particularly with sequential data A one-shot deep learning model was introduced to forecast the full degradation trajectory of lithium-ion batteries using only the first 100 cycles of data, effectively capturing capacity fade trends even under noisy conditions and offering a fast, scalable solution for early battery health prognostics (Li et al., 2021). An Independently Recurrent Neural Network (IndRNN)-based model was proposed for estimating the SoH of lithium-ion batteries in electric vehicles under variable load conditions utilizing voltage, current, and temperature data from NASA's randomized battery usage dataset; comparative analysis showed improved accuracy and robustness over Long Short-Term Memory (LSTM) and Gated Recurrent Units (GRUs) (Venugopal & Vigneswaran, 2019). LSTM networks have received significant attention and have been effectively applied to model temporal dependencies in battery degradation; an LSTM recurrent neural network was developed to predict remaining useful life (RUL), capturing long-term dependencies in degradation patterns and outperforming traditional machine learning models, thereby establishing a strong baseline for deep learning in battery health prognostics (Zhang, Xiong, He, & Pecht, 2018). A novel approach for estimating SoH using an improved LSTM integrated with carefully extracted health indicators—including charging time, capacity, and features from the incremental capacity curve—demonstrated strong generalization across datasets with fine-tuning-based transfer learning that reduced estimation errors, delivering robust and accurate SoH estimates across battery types and usage conditions (Ma, Shan, Gao, & Chen, 2022). An improved SoH estimation method that integrates a CNN–LSTM architecture with skip connections to address vanishing gradients and information loss, together with feature selection to eliminate redundant inputs, enhanced training efficiency (Xu et al., 2023). Finally, an SoH estimation method combining ridge regression and GRU to extract health indicators from voltage, current, and temperature data achieved RMSE and MAE within 1%, demonstrating stability and strong generalization in real-world energy storage conditions (Dai et al., 2024).

Furthermore, conventional deep learning models do not provide uncertainty estimates, which is a significant limitation when deploying such models in safety-critical applications like battery health management. Bayesian deep learning addresses this issue by modeling uncertainty in predictions, enhancing predictive reliability and enabling uncertainty aware decision making (Blundell, Cornebise, Kavukcuoglu, & Wierstra, 2015; Kendall & Gal, 2017). More recently, a Bayesian deep learning pipeline for SoH estimation with uncertainty quantification—arranged into modules for automatic feature selection and Bayesian neural network construction using Monte Carlo dropout—was proposed, demonstrating robust generalization across diverse battery datasets and positioning uncertainty-aware modelling as a benchmark for practical SoH estimation (Ke, Long, Yang, & Peng, 2024).

This study presents a Bayesian Convolutional Neural Network–Long Short-Term Memory (Bayesian CNN-LSTM) framework, augmented with a 'DenseVariational' layer, for the estimation of lithium-ion battery SoH with uncertainty quantification. A distinctive aspect of this work is the incorporation of a 'DenseVariational' layer within the CNN-LSTM architecture, enabling Bayesian inference at the dense layer level. This design facilitates uncertainty quantification in predictions while maintaining computational efficiency, addressing a critical limitation of conventional CNN-LSTM and other deep learning models for battery SoH estimation, which typically provide point estimates without associated uncertainty intervals. The absence of uncertainty measures in deterministic models can lead to overconfident predictions, posing risks in safety-critical applications such as battery management systems. By selectively applying variational inference, the proposed method achieves a balance between predictive accuracy and uncertainty quantification, providing informative uncertainty estimates under varying operating conditions.

To the best of our knowledge, Bayesian CNN–LSTM architectures have not yet been applied to battery SoH prediction, although they have shown promising results in other domains. The use of Bayesian CNN–LSTM for sentiment analysis in Massive Open Online Courses (MOOCs), integrating Bayesian optimization for hyperparameter tuning and achieving superior classification performance compared to traditional approaches, was first demonstrated in (Mrhar, Benhiba, Bourekkache, & Abik, 2021). Building on this concept, an advanced framework combining Bayesian CNN–LSTM with Deep Q-Learning as a meta-labeller for stock market trading strategies was introduced, enabling uncertainty-aware predictions and reinforcement learning-driven decision-making (Joel, 2023). More recently, a Bayesian optimization-based LSTM–CNN architecture for stock trend prediction was proposed, reversing the conventional integration order to prioritize temporal learning before spatial feature extraction. This model demonstrated significant improvements in accuracy, precision, and recall over standalone deep learning models (Chan et al., 2024). These studies collectively highlight the versatility and robustness of Bayesian CNN–LSTM architectures in handling uncertainty and improving predictive reliability across diverse applications, motivating their exploration for battery SoH estimation. In the battery prognostics context, variational inference is selectively applied in this work, at the dense layer of a CNN–LSTM architecture and posterior predictive intervals are evaluated on a randomized usage NASA dataset representative of real operating conditions.

The Bayesian CNN-LSTM model leverages the NASA random walk battery dataset, which features non-uniform charge-discharge cycles and irregular degradation patterns—closely mirroring real-world operating conditions often overlooked in conventional datasets. By incorporating Bayesian inference into the deep learning architecture, the framework enables probabilistic weight distributions and uncertainty quantification during inference, which is critical for safety-critical applications such as battery health monitoring. The hybrid architecture combines CNNs for spatial feature extraction from voltage, current, and time-series data, with LSTMs to capture temporal dependencies and degradation dynamics across cycles. This synergy improves feature representation and predictive accuracy while also providing posterior predictive intervals for SoH, thereby providing uncertainty information alongside SoH predictions.

To enrich the input features, we engineered parameters derived from the cycling data and employed a masking strategy to ensure a uniform input sequence length of 1500 time steps, even when fewer data points are available. The model is trained using random walk cycle data as input, with change in SoH during consequent cycles as the target, computed as normalized capacity measured during reference discharge cycles. The Bayesian formulation of the model supports improved generalization across different battery cells and operating conditions, making it potentially applicable to transfer learning and cross-cell prediction tasks. Comparative evaluations demonstrate that the proposed method achieves competitive performance relative to both traditional machine learning models (e.g., GPR, SVR, Beta regression) and deterministic deep learning approaches, while additionally providing uncertainty estimates.The architecture is also adaptable for online learning, allowing continual updates as new data becomes available—an essential capability for real-time SoH tracking applications.

The main contributions of this paper are:

- A Bayesian CNN-LSTM architecture designed using 'DenseVariational' layer for SoH prediction under random cycling conditions;
- An input pipeline with normalization, feature engineering, and masking to handle variable-length sequences;
- Integration of uncertainty estimation into the prediction process, yielding posterior predictive intervals to quantify prediction confidence;
- Empirical validation using a realistic NASA battery dataset, achieving an overall average R^2 score 0.932 of and RMSE score of 0.022.

The rest of this paper is organized as follows: the NASA randomized dataset used in the current work, preprocessing, model architecture, and training approach are discussed in Section 2. The evaluation strategy and results are discussed in Section 3. A detailed expositions of the key conclusions is provided in Section 4.

2. METHODOLOGY

The methodology used to develop a Bayesian CNN-LSTM model for time-series prediction is presented in this section. The model integrates Convolutional Neural Networks (CNNs) for feature extraction, Long Short-Term Memory (LSTM) networks for learning temporal dependencies, and a Bayesian framework for learning uncertainty in data. These components work in synergy to provide predictions.

2.1. Dataset

The main goal of this work is to develop a data-driven approach for estimating the state of health of a battery cell with good predictive accuracy for improved battery management. The proposed method is designed to handle variations in charge and discharge patterns observed in practical operating conditions. For this purpose, it is desirable to have a dataset which is generated using random usage of a battery that represents the real-world scenario, where load is not constant and charging profiles are random. One such dataset is the NASA randomized battery dataset (Bole, Kulkarni, & Daigle, 2014b) obtained from tests on a satellite battery under random load and charging profiles. The data was collected by subjecting the system to 7 different charging and discharging conditions. Four similar battery cells were used for each test condition. All three states of the battery, charging, discharging and rest, were realized randomly. Further, the cells were subjected to reference cycles in between, at a frequency of 500 or 1500 random cycles to measure the battery performance parameters such as battery health. An overview of these tests are summarized in Table 1.

In this dataset, three types of reference profiles were used, namely, 'low current discharge', 'reference charge and discharge cycle' and 'pulsed current discharge and charge'. The reference charge cycle starts when battery voltage is at its lower limit, 3.2 V, and ends when battery voltage is at its upper limit, 4.2V. These charge cycles follow industry standard constant current (CC) and constant voltage (CV) charging. Similarly, reference discharge starts when charging stops at 4.2V and ends when voltage reaches 3.2V. As mentioned earlier, between each of the reference cycles, battery cells were charged-discharged-rested using random load profiles, charge profiles and rest times. These are termed as 'Random Walk (RW) cycles' with load and operation times sampled from a probability distribution which can be seen in Table 1. Reference discharge cycles provide capacities used to compute the incremental change in SoH (dSoH), labels; RW cycles provide the input sequences. The final SoH trajectory is reconstructed by accumulating predicted $dSoH$ at test time. The SoH data was extracted from reference cycles. The time series profiles

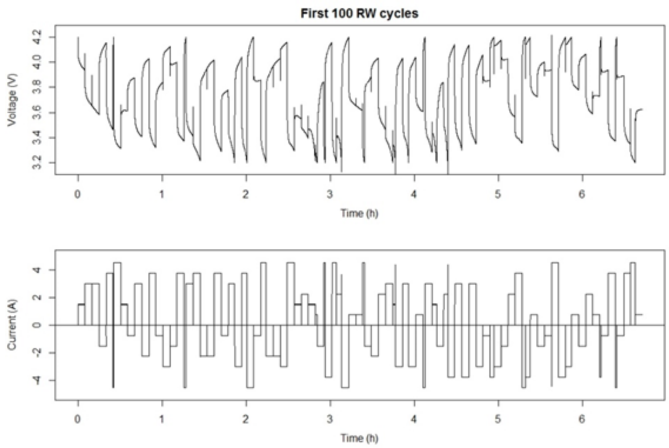

Figure 1. Profiles of voltage and current from a sample of first 100 random walk cycles

of current and voltage from one of the seven datasets from NASA randomized battery dataset are plotted in Fig. 1. Negative values of current represent charging current while positive values represent discharging current. When the current is zero battery is in rest. Rest time is very short, generally lasting less than a second in majority of cases.

The supervised target is the incremental SoH between consecutive reference–discharge cycles:

$$\mathrm{dSoH}_k = \mathrm{SoH}_k - \mathrm{SoH}_{k-1}, \tag{1}$$

where the absolute SoH at cycle k is given by

$$\mathrm{SoH}_k = \frac{C_k}{C_{\mathrm{ref}}}. \tag{2}$$

At inference, the absolute SoH is reconstructed by accumulating the predicted increments:

$$\widehat{\mathrm{SoH}}_k = \mathrm{SoH}_{k-1} + \widehat{\mathrm{dSoH}}_k. \tag{3}$$

Reference discharge cycles provide the ground-truth labels (capacities C_k), while RW cycles supply the input feature sequences.

To effectively utilize the randomized dataset for model training, appropriate preprocessing steps are required. The following subsection describes the data normalization techniques, feature engineering, and sequence preparation strategies employed.

2.2. Feature Creation and Data Processing

The first step involves preparing the raw data for the model by extracting meaningful features and ensuring uniform sequence lengths. The initial step involved the processing of the raw data to standardize its format for subsequent analysis. Several preprocessing steps are applied to derive the necessary features, which include data normalization, feature engineering, and padding. The key variables—current, voltage, and relative time—are normalized to standard ranges: current is rescaled to a range of $[-4.5, 4.5]$, voltage to $[3.2, 4.2]$, and relative time is scaled to $[0, 1]$ by dividing by 300. (300 seconds corresponds to the maximum duration of a RW cycle in the dataset; cycles may terminate earlier upon reaching voltage limits.) These normalized features are crucial for model stability and ensure that the input data is on a consistent scale. In addition to normalization, several derived features are created to capture important information about the system's performance:

a. 'Cycle Charge' is calculated by integrating current over time, which measures the total charge transferred during the cycle.
b. 'Duration of each cycle' is determined by subtracting the minimum relative time from the maximum relative time.
c. 'Throughput', 'Power', and 'Heat Loss' are derived by integrating their respective values over time during the cycle. These features capture the overall performance and efficiency of the system.

For clarity, the engineered quantities used in this work are defined as follows. The cycle–throughput is

$$Q_{\mathrm{cyc}} = \int_{t_0}^{t_1} |I(t)|\, dt \quad (\mathrm{A{\cdot}s}),$$

the accumulated power throughput is

$$P_{\mathrm{cyc}} = \int_{t_0}^{t_1} V(t)\, I(t)\, dt \quad (\mathrm{V{\cdot}s}),$$

and the heat–loss proxy is

$$H_{\mathrm{loss}} = \int_{t_0}^{t_1} I(t)^2\, dt \quad (\mathrm{A^2{\cdot}s}).$$

The duration of each charge–discharge cycle is

$$\Delta t = t_1 - t_0 \quad (\mathrm{s}).$$

Next, the dataset is filtered to focus on specific types of cycles:

a. "Reference Discharge Cycles" are selected as the benchmark for comparison.
b. "RW Cycles" (both "discharge" and "charge") are also included for feature extraction, which involves throughput, power, heat loss, and current/voltage values.

After feature extraction, the dataset is padded to ensure uniform sequence lengths. Padding is added to shorter sequences to make them of equal length, ensuring consistency across all input data for model training. The supervised target used to train the network is $dSoH$ (defined in 2.1), obtained from capacities measured at consecutive reference discharge cycles; the absolute SoH trajectory is reconstructed at inference by accumulation ($SoH_{k1} + dSoH_k$).

Table 1. Summary of NASA randomized battery dataset (Bole et al., 2014b)

Dataset No.	Dataset Title	Battery code	Random Walk (RW) Cycle Description
1	Battery Uniform Distribution Charge Discharge DataSet 2Post	RW9, RW10, RW11, RW12	Random charge or discharge with current 0.75A, 1.5A, 2.25A, 3A, 3.75A, or 4.5A. Charging continues until battery voltage reaches 4.2V, or 5 minutes have passed. Discharging continues until battery voltage drops to 3.2V, or 5 minutes have passed.
2	Battery Uniform Distribution Discharge Room Temp DataSet 2Post	RW3, RW4, RW5, RW6	Constant-current charging at 2A until 4.2V, followed by constant-voltage charging until current drops below a predefined threshold. Random discharge with currents between 0.5A and 4A. Discharging continues until battery voltage drops to 3.2V, or 5 minutes have passed. A new discharging setpoint is selected with a <1s rest period.
3	Battery Uniform Distribution Variable Charge Room Temp DataSet 2Post	RW1, RW2, RW7, RW8	Battery is charged randomly for 0.5h, 1h, 1.5h, 2h, 2.5h, or until full. Additionally, batteries are charged at 2A (CC then CV) until either the battery reaches 4.2V or the charging time exceeds the allotted interval. Random discharge with currents between 0.5A and 4A. Discharging continues until battery voltage drops to 3.2V, or 5 minutes have passed. A new discharging setpoint is selected with a <1s rest period.
4	RW Skewed High 40°C DataSet 2Post	RW13–RW28	Constant-current charge at 2A until 4.2V, then constant-voltage charge until current drops below a predefined threshold. Random discharge with currents between 0.5A and 4A. Discharging continues until battery voltage drops to 3.2V, or 5 minutes have passed. A new discharging setpoint is selected with a <1s rest period.
5	RW Skewed High Room Temp DataSet 2Post		
6	RW Skewed Low 40°C DataSet 2Post		
7	RW Skewed Low Room Temp DataSet 2Post		

The model processes four engineered features per cycle, resulting in a sequential input representation of size (T, 4), where T denotes the number of Random Walk (RW) cycles between consecutive reference cycles. Specifically, each RW cycle is represented by a feature vector $x_t = [Q_{cyc}, P_{cyc}, H_{loss}, \Delta t]$, where the features are computed by integrating current and voltage signals over the duration of a single cycle. Thus, the input to the model is a sequence of cycle-level feature vectors rather than raw time-series measurements. Across the dataset, up to 1500 RW cycles occur between successive reference measurements, resulting in input matrices of size up to 1500 × 4 for each training instance. Each such sequence is associated with a single target value corresponding to the change in SoH (dSoH) measured at the subsequent reference discharge cycle, all of which are normalized prior to training. Input sequences are padded and masked to a fixed length of 1500 time steps.

2.3. Theoretical Background

Bayesian neural networks (BNNs) are employed for integrating uncertainty into the predictions by treating the parameters of the network as random variables with distributions. While traditional neural networks operate with static parameters, BNNs define a prior distribution that reflects initial beliefs concerning model parameters, serving as a foundation for uncertainty estimation even before data observation. As new data becomes available, the posterior distribution is iteratively refined, providing the continually updated representation of parameter uncertainty given the accumulating evidence. The model incorporates uncertainty propagation to quantify the uncertainty associated with its predictions.

For the present case, the prior distribution for each parameter is modeled as a zero-mean Gaussian with a fixed standard deviation of 0.01, reflecting a regularizing prior consistent with the implemented ‘DenseVariational’ layer. The posterior distribution, which is learned through variational inference, is also modeled as a Normal distribution, with the mean (loc) and scale (scale) as trainable parameters. Variational inference allows for an efficient approximation of the posterior by optimizing the parameters of a simpler distribution that minimizes the Kullback-Leibler (KL) divergence to the true posterior.

A zero–mean Gaussian prior over the weights is adopted,

$$p(\mathbf{w}) = \mathcal{N}\big(\mathbf{0},\, 0.01^2\,\mathbf{I}\big)\,.$$

The variational posterior is a mean–field Gaussian with a softplus scale,

$$q_\theta(\mathbf{w}) = \mathcal{N}\big(\boldsymbol{\mu},\, \boldsymbol{\sigma}^2\mathbf{I}\big)\,, \qquad \boldsymbol{\sigma} = \mathrm{softplus}(\mathbf{s}),$$

where $(\boldsymbol{\mu}, \mathbf{s})$ are learned per–weight parameters.

The variational training objective combines a data–fit term with a complexity penalty:

$$\mathcal{L} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \;+\; \beta \cdot \mathrm{KL}(q_\theta(\mathbf{w}) \,\|\, p(\mathbf{w}))\,.$$

Here, the Mean Squared Error (MSE) acts as a Gaussian-likelihood surrogate, and $\mathrm{KL}(q\|p)$ regularizes the variational posterior toward the prior. (Implementation details for β are provided in 2.5.)

The model also integrates CNNs to capture local temporal features and LSTM networks to model long-range temporal dependencies within the transformed feature representation. CNNs, typically used in image processing, are effective in time-series data for detecting local patterns or temporal features, while LSTMs excel at learning from sequences and modeling dependencies within the transformed sequential feature representation. The model can learn both local and long-term temporal patterns effectively by combining CNNs and LSTMs.

The ‘DenseVariational’ layer captures epistemic (model) uncertainty by placing distributions over the network weights. In contrast, aleatoric (data) uncertainty is not explicitly modeled, as the network outputs only the mean prediction for the supervised target.

2.4. Model Architecture

The model architecture integrates several key components for processing time-series data, extracting salient features, learning temporal dependencies, and quantifying predictive uncertainty. Although the original data consists of high-frequency voltage and current measurements, these are first aggregated at the cycle level to construct a compact and consistent sequential representation. Consequently, the model operates on sequences of cycle-level feature vectors rather than raw time-series signals. The architecture is as follows:

i. *Input Layer:* The model processes time-series datasets structured as a three-dimensional tensor (N, T, F). In this representation, N denotes the total number of samples, T specifies the number of sequential time steps within each sample, and F represents the number of distinct features observed at each time step. All input sequences are padded to a fixed length of 1500 time steps to ensure consistent input dimensions across cells.

ii. *Masking Layer:* A masking layer is applied so that padded (zero-valued) time steps are ignored by all downstream layers. This ensures that the model is not misled by incomplete or irrelevant data points.

iii. *Convolutional Layer:* 1D convolutional layer comprising 64 filters, each with a kernel size of 3 is used to process the input data. This layer detects local temporal features in the input sequence. The convolution operation scans through the data, applying filters that highlight important temporal patterns or trends.

iv. *Global Average Pooling:* After the convolutional layer, global average pooling is applied to reduce the dimensionality of the feature map obtained in previous step. This operation computes the mean value across each feature map, summarizing the most relevant information while keeping the output compact. The convolutional layer first extracts localized temporal patterns across the sequence of cycle-level features. The Global Average Pooling operation then aggregates these responses across the time dimension, effectively summarizing information from multiple neighboring time steps into a compact feature representation. Consequently, the temporal structure is not entirely removed but is encoded within the convolutional feature maps. The subsequent LSTM layer operates on this condensed representation to model dependencies within the transformed feature space rather than directly across the original long sequence.

v. *LSTM Layer:* After the global average pooling layer, the network architecture includes an LSTM layer with 32 units is applied to capture dependencies within the transformed feature representation derived from the input sequence. This layer processes input sequences and learns patterns across time steps, enabling the model to effectively represent dynamic relationships in the dataset. By leveraging its recurrent structure, the LSTM layer enhances predictive accuracy for degradation trend.

vi. *Reshaping for LSTM:* The pooled output of the convolutional layer is restructured into a 3D tensor suitable for input to the LSTM layer. This reshaping prepares the pooled feature representation for sequential processing by the LSTM layer.

vii. *DenseVariational layer:* The ‘DenseVariational’ layer, configured with 16 units, introduces Bayesian inference into the model by learning a distribution over its weights instead of fixed values. This component serves as an alternative to the conventional dense layer, enabling the model to quantify the uncertainty associated with its predictions. To achieve this, a prior distribution over the weights is defined, and a corresponding posterior distribution is subsequently learned through variational inference. During training, the model samples from this posterior to make probabilistic predictions. This methodology enhances model stability and concurrently decreases the likelihood of overfitting.

viii. *Fully Connected Layers:* Following the LSTM layer, a single fully connected dense layer with 1 unit is incorporated into the network architecture. This layer outputs a single value representing the model’s prediction. The network outputs a point prediction per forward pass; predictive intervals are obtained by sampling multiple weight realizations from the variational posterior (see 2.6).

Fig. 2 is a schematic diagram of the overall architecture and layer configuration used in the Bayesian CNN-LSTM model.

2.5. Model Training

The proposed Bayesian CNN–LSTM model is trained using the ‘Adam’ optimizer with a MSE loss, which is appropriate

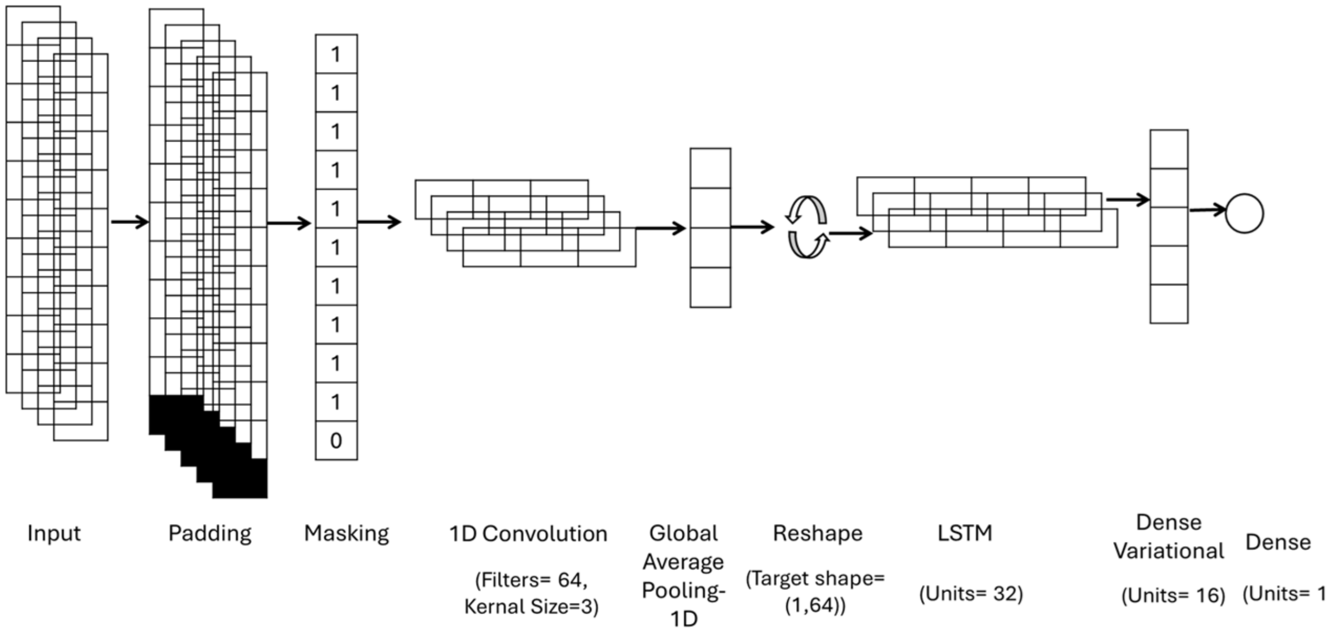

Figure 2. Architecture of Bayesian CNN-LSTM model

for regression tasks such as SoH estimation. In addition to the data-fit term, the training objective includes a Kullback–Leibler (KL) divergence regularizer applied within the 'DenseVariational' layer. The KL contribution is scaled using a weighting factor equal to the inverse of the minibatch size,

$$\text{KLweight} = \frac{1}{N},$$

corresponding to $\beta = 1/N$ in the variational objective (see 2.3). This scaling normalizes the KL term per batch. No KL warm-up schedule is employed; the weighting factor β remains constant throughout training. Training is performed for 50 epochs with a batch size of 8.

During training, the parameters of the convolutional, LSTM, and dense layers are updated via backpropagation. The variational inference mechanism enables the 'DenseVariational' layer to learn a posterior distribution over its weights, providing an estimate of epistemic uncertainty.

A formal hyperparameter search (grid, random, or Bayesian optimization) is not conducted in this study. Instead, hyperparameters are tuned manually using single-parameter adjustments to reduce validation error on the training cells. No explicit search ranges are enforced, and the same final configuration is used across all seven randomized usage conditions. The final hyperparameters applied in all experiments are summarized in Table 2

Leave-one-out cross validation (LOOCV) across cells is employed only for performance reporting. Hyperparameters are fixed prior to LOOCV and are not updated using information from the held-out cell. After training, the resulting model configuration is evaluated on unseen battery data to assess its ability to predict SoH and quantify predictive uncertainty under randomized operating conditions.

Table 2. Final hyperparameter configuration used in all experiments.

Hyperparameter	Final value	How chosen
Optimizer	Adam	fixed
Learning rate	1×10^{-3}	manual tuning
Batch size	8	fixed
Epochs	50	fixed
Conv1D filters / kernel	64 / 3	architecture choice
LSTM units	32	manual tuning
DenseVariational units	16	manual tuning
KL weight β	$1/N$	fixed
Random seed	72	fixed
Inference samples S	100	fixed

2.6. Uncertainty Estimation

Predictive uncertainty: Predictive uncertainty is estimated by sampling the variational weights $\mathbf{w}^{(s)} \sim q_\theta(\mathbf{w})$ for $s = 1, \ldots, S$ in the 'DenseVariational' layer, and performing forward passes

$$\hat{y}^{(s)} = f(\mathbf{X};\, \mathbf{w}^{(s)}).$$

The predictive mean is computed as

$$\bar{y} = \frac{1}{S} \sum_{s=1}^{S} \hat{y}^{(s)}.$$

Predictive intervals: Posterior predictive intervals are reported using a Gaussian approximation,

$$\bar{y} \pm 1.64\,\sigma,$$

corresponding to a nominal 90% interval, where σ is the standard deviation of the set $\{\hat{y}^{(s)}\}_{s=1}^{S}$. Unless stated otherwise, $S = 100$ samples are used in all experiments.

Uncertainty metrics: Evaluation includes standard probabilistic and interval-based metrics- Prediction Interval Coverage Probability (PICP) and Mean Prediction Interval Width (MPIW) at 90% and 95%, together with Negative Log- Likelihood (NLL) and Continuous Ranked Probability Score (CRPS), which quantify predictive calibration and overall probabilistic accuracy. The reported NLL values correspond to the Gaussian negative log-likelihood and may take negative values due to the logarithmic term in the likelihood, particularly when the predictive variance is small and the target variable is normalized in the range $[0, 1]$.

3. RESULTS AND DISCUSSION

The evaluation framework and predictive outcomes of the proposed Bayesian CNN-LSTM model for battery SoH estimation are presented in this section. The model's performance is analyzed using standard metrics and discussed in the context of realistic battery usage patterns.

The SoH model was trained on a workstation having the processor specifications: 'Intel(R) Core (TM) i5-8365U CPU@1.60-GHz, 1.90 GHz' with installed RAM capacity of 8.00 GB. The model was developed in Python 3.11, leveraging the TensorFlow, TensorFlow Probabilty (TFP) and Keras libraries for its implementation. Data preprocessing, model training, and evaluation were performed on this setup.

$S = 100$ stochastic forward passes through the 'DenseVariational' layer were drawn for each sequence and the per–time–step predictive mean (μ) and standard deviation (σ) were computed $\mu \pm 1.64\,\sigma$

3.1. Evaluation Metrics

In addition to point accuracy, uncertainty–quality metrics are reported to assess the calibration and reliability of the predictive distributions. The evaluation metrics considered in the present work are 'Coefficient of Determination', popularly known as R^2, and 'Normalized Root Mean Squared Error ($RMSE_{norm}$)' to evaluate and compare the results.

a. **Coefficient of Determination (R^2):** This statistical measure quantifies the model's fidelity to the actual data points. It has the upper limit of 1, which signifies that model exactly fits real data points.

$$\begin{aligned} R^2 &= \frac{SS_{reg}}{SS_{total}} \\ &= 1 - \frac{SS_{res}}{SS_{total}} \end{aligned} \tag{4}$$

where, SS_{reg} = Sum of Squared error explained by Regression, SS_{res} = Sum of Squared error or Residuals, and SS_{total} = Total Sum of Squares.

$$\begin{aligned} SS_{reg} &= \sum_{i=1}^{N} (\hat{C}_i - \bar{C}_i)^2 \\ SS_{res} &= \sum_{i=1}^{N} (C_i - \hat{C}_i)^2 \\ SS_{total} &= \sum_{i=1}^{N} (C_i - \bar{C}_i)^2 \end{aligned} \tag{5}$$

Where, N = total number of cycles, $\hat{C}_i$ = predicted capacity at i^{th} cycle, $\bar{C}_i$ = mean capacity, C_i = Actual Capacity at i^{th} cycle.

b. **Normalized Root Mean Squared Error ($RMSE_{norm}$):** It provides interpretable measure of the scaled average magnitude of prediction error with respect to target variable, making it particularly useful for assessing the model's performance. Lower $RMSE_{norm}$ values indicate better alignment between estimated and actual SoH values.

$$RMSE_{norm} = \sqrt{\frac{1}{N} \sum_{i=1}^{n} (\frac{\hat{C}_i - C_i}{C_i})^2} \tag{6}$$

Additional uncertainty metrics: The evaluation includes: (i) Prediction Interval Coverage Probability (PICP) at 90% and 95% (empirical fraction of targets captured by the predictive interval), (ii) Mean Prediction Interval Width (MPIW) at 90% and 95%, (iii) Gaussian Negative Log-Likelihood (NLL), and (iv) Continuous Ranked Probability Score (CRPS). These metrics complement the pointwise measures R^2 and $\mathrm{RMSE}_{\mathrm{norm}}$ by quantifying interval calibration and distributional accuracy. The relatively low (negative) NLL values observed in this study indicate that the model assigns high likelihood to the observed targets, reflecting a good probabilistic fit under normalized target scaling.

3.2. State of Health Estimation

In the NASA randomized battery usage dataset, each condition contains four cells. For each condition, we adopt leave one cell out cross validation (LOOCV): train on three cells and evaluate on the held out cell, rotating across the four folds. Hyperparameters are fixed from the training cells only (as stated in 2.5); LOOCV is used strictly for performance reporting. A set of baseline models is considered for comparison, including deep learning approaches such as LSTM, GRU, deterministic CNN–LSTM,and a non-Bayesian dense-output variant as 'Dense Counterpart' for convenience (the same architecture with dense layer replacing 'DenseVariational' layer), as well as traditional machine learning models such as Beta regression, Gaussian Process Regression (GPR), and Support Vector Regression (SVR). Condition-wise and averaged point-accuracy results for all models are presented in Tables 3 & 4. For the Bayesian CNN–LSTM model, predictive uncertainty is quantified using PICP (90%, 95%), MPIW (90%, 95%), NLL, and CRPS, with results summarized in Table 5. The observed coverage values are slightly below the nominal levels (90% and 95%), indicating mild under-coverage and suggesting that the model exhibits slight overconfidence in certain conditions. This behavior can be attributed to the limited number of available reference cycles, as capacity measurements are obtained only intermittently (after long random-walk sequences), resulting in relatively few supervised target points for calibration. Despite this limitation, the predictive intervals capture the overall degradation trends effectively and provide meaningful uncertainty estimates across diverse operating conditions. These metrics assess interval coverage and distributional accuracy in addition to point-wise error. The results show that the proposed Bayesian CNN–LSTM model performs competitively across both traditional and deep learning models under varying operating conditions.

The performance metrics reported in Tables 3 & 4 are the average values obtained for all 7 test conditions. A comparison with deterministic baselines indicates that the proposed Bayesian CNN–LSTM achieves comparable point prediction accuracy to its deterministic counterparts. In several cases, the deterministic CNN–LSTM or dense counterpart performs similarly, and occasionally slightly better. This indicates that the primary advantage of the Bayesian formulation lies not in improved point accuracy, but in its ability to provide uncertainty estimates that support qualitative assessment of prediction confidence.

Similarly, comparisons with traditional machine learning models indicate that while methods such as GPR and Beta regression perform well under certain conditions, the proposed approach maintains consistent performance while additionally offering uncertainty quantification.

Model predictions obtained using Bayesian CNN-LSTM model for RW1, RW9, RW14 and RW26 cells are shown in Fig. 3, 4, 5 and 6 respectively. The model captures the trend of capacity degradation as well as capacity increase after long rest reasonably well. The shaded regions in Figures 3–6 denote approximate 90% posterior predictive intervals, computed as $\mu \pm 1.64\,\sigma$ using $S = 100$ stochastic forward passes through the variational layer (see 2.6)

Training time per epoch is approx.707ms on average. It should

Table 3. Coefficient of determination (R^2) comparison across models.

Battery Pack Name	**Bayesian CNN-LSTM**	**LSTM**	**GRU**	**CNN-LSTM**	**Dense Counterpart**	**Beta Regression**	**Gaussian Process (Matérn Kernel)**	**Support Vector Regression**
Battery Uniform Distribution Charge Discharge DataSet 2Post	0.986	0.964	0.961	0.961	0.988	0.860	0.891	0.825
Battery Uniform Distribution Discharge Room Temp DataSet 2Pos	0.954	0.956	0.955	0.947	0.948	0.673	0.771	0.723
Battery Uniform Distribution Variable Charge Room Temp DataSet 2Post	0.961	0.887	0.885	0.872	0.963	0.874	0.869	0.873
RW Skewed High 40C DataSet 2Post	0.890	0.898	0.883	0.926	0.887	0.700	0.727	0.741
RW Skewed High Room Temp DataSet 2Post	0.954	-0.304	-0.339	0.961	0.940	0.930	0.907	0.750
RW Skewed Low 40C DataSet 2Post	0.888	0.892	0.866	0.841	0.873	0.823	0.512	0.736
RW Skewed Low Room Temp DataSet 2Post	0.890	0.944	0.943	0.883	0.887	0.947	0.914	0.933
Groupwise Average	**0.932**	0.748	0.736	0.913	0.927	0.830	0.799	0.806

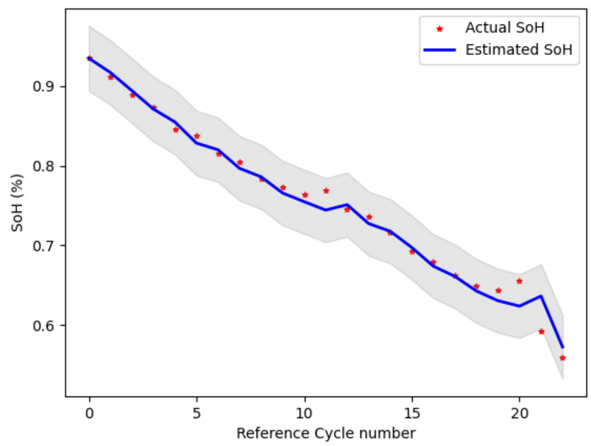

Figure 3. Comparison of predicted and measured SoH for test battery cell RW1. Shaded regions denote approximate 90% posterior predictive intervals, computed as $\mu \pm 1.64\,\sigma$ from $S = 100$ stochastic forward passes.

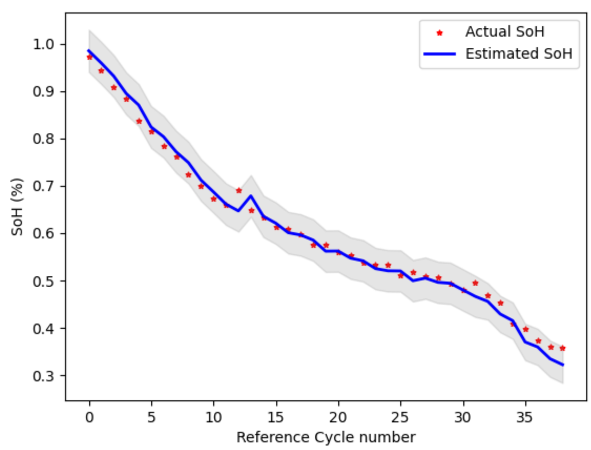

Figure 4. Comparison of predicted and measured SoH for test battery cell RW9. Shaded regions denote approximate 90% posterior predictive intervals, computed as $\mu \pm 1.64\,\sigma$ from $S = 100$ stochastic forward passes.

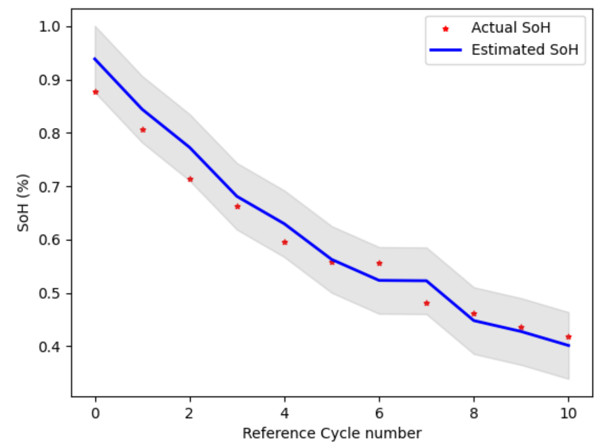

Figure 5. Comparison of predicted and measured SoH for test battery cell RW14. Shaded regions denote approximate 90% posterior predictive intervals, computed as $\mu \pm 1.64\,\sigma$ from $S = 100$ stochastic forward passes.

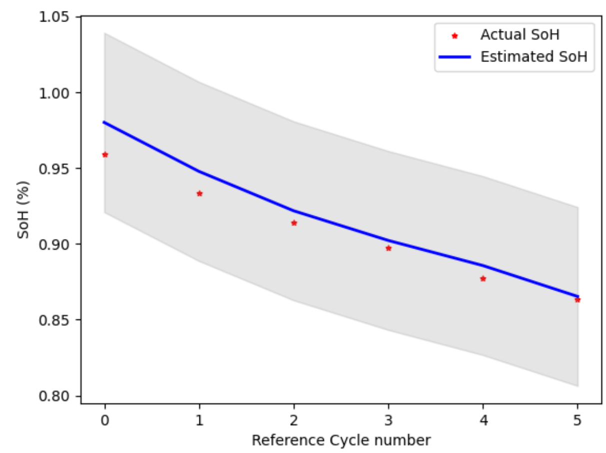

Figure 6. Comparison of predicted and measured SoH for test battery cell RW26. Shaded regions denote approximate 90% posterior predictive intervals, computed as $\mu \pm 1.64\,\sigma$ from $S = 100$ stochastic forward passes.

be noted that the Bayesian CNN-LSTM network provides accurate SoH estimation with consistent performance, along with uncertainty estimates exhibiting reasonable empirical coverage across operating conditions. However, it does come with certain limitations. The method is computationally more intensive than standard LSTM models due to the sampling-based inference needed to estimate uncertainty. This leads to increased computational time for both model training and inference. Over-regularization from the Bayesian framework may lead to underfitting, especially with limited data. Another limitation of this study is the exclusion of temperature's impact on SoH. It is crucial to note that real-world conditions necessitate consideration of this factor. Incorporating temperature and other environmental parameters in future studies may further enhance the accuracy and applicability of SoH prediction models across diverse operating conditions. Despite these drawbacks, the benefits in accuracy and uncertainty quantification make it a compelling choice for real-world battery health prediction.

Deterministic models achieve comparable point-prediction performance, indicating that the primary advantage of the Bayesian formulation lies in uncertainty estimation rather than accuracy improvement.

4. CONCLUSION AND FUTURE WORK

Accurate prediction of the SoH of batteries is essential for the efficient operation of electric vehicles, particularly under real-world conditions where charging and discharging occur in a random manner. In this study, a Bayesian CNN-LSTM network was employed to estimate SoH, offering an accurate solution with potential applicability in battery management workflows, subject to further validation. By leveraging the sequential learning capability of LSTM networks along with CNNs and Bayesian inference, the model effectively captures temporal dependencies in battery behavior while quantifying uncertainty in predictions. This makes the approach not only accurate but also provides uncertainty estimates under varying loading conditions. In addition to the visual intervals, uncertainty was evaluated quantitatively in section– 3 using interval-based and probabilistic metrics.

In addition to point accuracy, the Bayesian formulation provides reasonably calibrated predictive intervals with acceptable empirical coverage (as defined in 2.6), supporting risk aware assessment of estimation confidence under diverse operating conditions. Moreover, the Bayesian CNN-LSTM network exhibits strong generalization capabilities across randomized charge-discharge profiles, which are common in practical applications. The ability of the model to handle noise and variability in the data suggests potential applicability in battery management applications. Compared to conventional approaches, the proposed model achieves performance that is competitive with both traditional machine learning and deterministic deep learning methods, while additionally providing uncertainty-aware predictions. This highlights the value of incorporating Bayesian inference primarily for uncertainty quantification rather than for improving point-prediction accuracy.

4.1. Future Work

While the proposed approach demonstrates promising results, several avenues remain for future research:

- *Integration of Real-Time Temperature Data:* Current datasets are generated under controlled laboratory conditions, which do not fully capture thermal variations in real-world environments. Incorporating real-time temperature measurements can improve model robustness and accuracy. In particular, co training with temperature channels from on board sensors and including temperature aware engineered features (e.g., moving window thermal statistics) are expected to improve $dSoH$ prediction under realistic duty cycles.
- *Physics-Informed Modelling:* Deep learning models, though highly accurate, often require architecture or parameter adjustments for different battery specifications. They do not inherently represent the underlying electrochemical processes, limiting physical interpretability. Future work should explore hybrid approaches combining data-driven and physics-based models.
- *Model Interpretability:* Deep learning models are inherently black-box in nature, making interpretation challenging. Developing explainable AI techniques to identify key features influencing predictions and uncertainty will enhance trust and usability.
- *Aleatoric Uncertainty:* The current model captures epistemic uncertainty via weight distributions; extending the architecture to model aleatoric noise (e.g., a variance head for heteroscedastic regression) is a promising direction to improve total uncertainty decomposition.
- *Scalability and Deployment:* Extending the framework for real-time implementation in Battery Management Systems (BMS), including online learning and continual updates, remains a critical step for practical adoption.
- *Cross-Chemistry Generalization:* Investigating transfer learning strategies to adapt the model for different battery chemistries, such as solid-state or next-generation lithium-ion variants, will broaden applicability.

REFERENCES

Blundell, C., Cornebise, J., Kavukcuoglu, K., & Wierstra, D. (2015). Weight uncertainty in neural networks. In *Proceedings of the international conference on machine learning (icml)* (pp. 1613–1622).

Bole, B., Kulkarni, C., & Daigle, M. (2014b). *Randomized battery usage data set.* NASA Prognostics Data Repository, NASA Ames Research Center, Moffett Field, CA.

Bole, B., Kulkarni, C. S., & Daigle, M. (2014a). Adaptation of an electrochemistry-based li-ion battery model to account for deterioration observed under randomized use. In *Annual conference of the prognostics and health management society.*

Chan, B. K., Johnson, O. V., Chew, X., Khaw, K. W., Lee, M. H., & Alnoor, A. (2024). Proposed bayesian optimization based lstm-cnn model for stock trend prediction. *Computing and Informatics*, *43*(1), 38–63. doi: 10.31577/cai$_2$024$_1$$_3$8

Cribari-Neto, F., & Zeileis, A. (n.d.). *Beta regression in r.*

Dai, Z., Li, A., Sun, W., Zhou, H., Rao, R., & Luo, Q. (2024). Estimation and prediction method of lithium battery state of health based on ridge regression and gated recurrent unit. *IET Energy Systems Integration*.

Dong, G., Chen, Z., Wei, J., & Ling, Q. (2018). Battery health prognosis using brownian motion modelling and particle filtering. *IEEE Transactions on Industrial Electronics*, *65*(11).

Dubarry, M., Svoboda, V., Hwu, R., & Liaw, B. Y. (2008). Incremental capacity analysis and close-to-equilibrium ocv measurements to quantify capacity fade in commercial rechargeable lithium batteries. *Electrochimica Acta*, *53*(26), 9678–9684.

Ebden, M. (2015). *Gaussian processes: A quick introduction.*

Ferrari, S. L. P. (2013, February). *Beta regression modelling: recent advances in theory and applications.*

Ferrari, S. L. P., & Cribari-Neto, F. (2004). Beta regression for modelling rates and proportions. *Journal of Applied Statistics*, *31*(7).

Goebel, K., Saha, B., Saxena, A., Celaya, J. R., & Christophersen, J. P. (2008). Prognostics in battery health management. *IEEE Instrumentation and Measurement Magazine*, *11*(4).

Jaguemont, J., Boulon, L., & Dube, Y. (2016). A comprehensive review of lithium-ion batteries used in hybrid and electric vehicles at cold temperatures. *Applied Energy*, *164*, 99–114.

Joel, A. D. (2023). *Predicting implicit patterns and optimizing market entry and exit decisions in stock prices using integrated bayesian cnn-lstm with deep q-learning as a meta-labeller.* [GitHub Repository]. (Available at: https://github.com/adrikodebo/bayesian-CNN-LSTM-Q-learning)

Ke, Y., Long, M., Yang, F., & Peng, W. (2024). A bayesian deep learning pipeline for lithium-ion battery soh estimation with uncertainty quantification. *Quality and Reliability Engineering International*, *40*, 406–427.

Kendall, A., & Gal, Y. (2017). What uncertainties do we need in bayesian deep learning for computer vision? In *Proceedings of neurips* (pp. 5574–5584).

Li, W., Sengupta, N., Dechent, P., Howey, D., Annaswamy, A., & Sauer, D. U. (2021). One-shot battery degradation trajectory prediction with deep learning. *Journal of Power Sources*, *506*, 230024.

Ma, Y., Shan, C., Gao, J., & Chen, H. (2022). A novel method for state of health estimation of lithium-ion batteries based on improved lstm and health indicators extraction. *Energy*, *251*, 123973.

Mrhar, K., Benhiba, L., Bourekkache, S., & Abik, M. (2021). A bayesian cnn-lstm model for sentiment analysis in massive open online courses moocs. *International Journal of Emerging Technologies in Learning (iJET)*, *16*(23), 4–18.

Nagulapati, V. M., Lee, H., Jung, D., Sargunan, S., Paramanantham, S., Brigljevic, B., . . . Lim, H. (2021). A novel combined multi-battery dataset based approach for enhanced prediction accuracy of data driven prognostic models in capacity estimation of lithium ion batteries. *Energy and AI*, *5*, 100089.

Ramadass, P., Haran, B., White, R., & Popov, B. N. (2003). Mathematical modeling of the capacity fade of li-ion cells. *Journal of Power Sources*, *123*(2), 230–240.

Rasmussen, C. E. (2006). *Gaussian processes for machine learning*. CiteSeer.

Reitz, R. D., Ogawa, H., Payri, R., Fansler, T., Kokjohn, S., Moriyoshi, Y., . . . Zhao, H. (2019). The future of the internal combustion engine. *International Journal of Engine Research.*

Richardson, R. R., Osborne, M. A., & Howey, D. A. (2017). Gaussian process regression for forecasting battery state of health. *Journal of Power Sources*, *357*, 209–219.

Richardson, R. R., Osborne, M. A., & Howey, D. A. (2019). Battery health prediction under generalized condition using a gaussian process transition model. *Journal of Energy Storage*, *23*, 320–328.

Saha, B., & Goebel, K. (n.d.). *Battery data set.* NASA Ames Prognostics Data Repository.

Saha, B., Goebel, K., Poll, S., & Christophersen, J. (2009). Prognostics methods for battery health monitoring using a bayesian framework. *IEEE Transactions on Instrumentation and Measurement*, *58*(2), 291–296.

Saxena, A., Celaya, J., Balaban, E., Goebel, K., Saha, B., Saha, S., & Schwabacher, M. (2008). Metrics for evaluating performance of prognostic techniques. In *Ieee xplore.* Denver, CO, USA.

Venugopal, P., & Vigneswaran, T. (2019). State-of-health estimation of li-ion batteries in electric vehicle using indrnn under variable load condition. *Energies*, *12*, 4338.

Wang, D., Miao, Q., & Pecht, M. (2013). Prognostics of lithium-ion batteries based on relevance vectors and a conditional three-parameter capacity degradation model. *Journal of Power Sources*, *239*, 253–264.

Wei, J., Dong, G., & Chen, Z. (2018). Remaining useful life prediction and state of health diagnosis for lithium-ion batteries using particle filter and support vector regression. *IEEE Transactions on Industrial Electronics*, *65*(7).

Xu, H., Wu, L., Xiong, S., Li, W., Garg, A., & Gao, L. (2023). An improved cnn-lstm model-based state-of-health estimation approach for lithium-ion batteries. *Energy*, *276*, 127585.

Zhang, Y., Xiong, R., He, H., & Pecht, M. G. (2018). Long short-term memory recurrent neural network for remaining useful life prediction of lithium-ion batteries. *IEEE Transactions on Vehicular Technology*, *67*(7), 5695–5705.

Table 4. Normalized root mean squared error (RMSE$_{norm}$) comparison across models.

Battery Pack Name	**Bayesian CNN-LSTM**	**LSTM**	**GRU**	**CNN-LSTM**	**Dense Counterpart**	**Beta Regression**	**Gaussian Process (Matérn Kernel)**	**Support Vector Regression**
Battery Uniform Distribution Charge Discharge DataSet 2Post	0.014	0.026	0.027	0.016	0.014	0.085	0.078	0.084
Battery Uniform Distribution Discharge Room Temp DataSet 2Pos	0.020	0.020	0.020	0.023	0.021	0.095	0.077	0.089
Battery Uniform Distribution Variable Charge Room Temp DataSet 2Post	0.020	0.029	0.029	0.028	0.020	0.061	0.063	0.063
RW Skewed High 40C DataSet 2Post	0.025	0.012	0.013	0.027	0.025	0.026	0.041	0.037
RW Skewed High Room Temp DataSet 2Post	0.020	0.119	0.123	0.019	0.021	0.055	0.058	0.058
RW Skewed Low 40C DataSet 2Post	0.027	0.022	0.024	0.029	0.029	0.030	0.044	0.041
RW Skewed Low Room Temp DataSet 2Post	0.025	0.030	0.031	0.027	0.025	0.050	0.061	0.058
Groupwise Average	**0.022**	0.037	0.038	0.024	0.022	0.057	0.060	0.061

Table 5. Interval reliability and distributional accuracy for the Bayesian CNN–LSTM model.

Battery Pack Name	**PICP** (90%)	**MPIW** (90%)	**PICP** (95%)	**MPIW** (95%)	**CRPS**	**NLL**
Battery Uniform Distribution Charge Discharge DataSet 2Post	0.897	0.073	0.938	0.088	0.014	-2.394
Battery Uniform Distribution Discharge Room Temp DataSet 2Pos	0.894	0.059	0.928	0.070	0.015	-2.511
Battery Uniform Distribution Variable Charge Room Temp DataSet 2Post	0.904	0.089	0.945	0.107	0.016	-2.385
RW Skewed High 40C DataSet 2Post	0.879	0.024	0.918	0.028	0.006	-3.489
RW Skewed High Room Temp DataSet 2Post	0.851	0.296	0.898	0.406	0.055	-0.918
RW Skewed Low 40C DataSet 2Post	0.783	0.049	0.883	0.058	0.018	-2.325
RW Skewed Low Room Temp DataSet 2Post	0.873	0.076	0.921	0.091	0.018	-2.222
Groupwise Average	**0.855**	**0.095**	**0.895**	**0.121**	**0.020**	**-2.321**