Estimating the Battery State of Health with Quantified Aleatoric and Epistemic Uncertainty

Joshua Bogaert[1], Ingeborg de Pater[2]

[1,2] *Delft University of Technology, Faculty of Aerospace Engineering, Delft, 2629HS, the Netherlands*
[2] *i.i.depater@tudelft.nl*

Abstract

Batteries are crucial in the transition towards a sustainable society. There is therefore an increased interest in battery health management. In battery health management, one of the key quantities is the State of Health (SoH), i.e., the maximum capacity of a battery, which decreases over time as the battery degrades. Accurate SoH estimations are needed to plan operations and battery replacements.

A crucial challenge in SoH estimation is to quantify the uncertainty of the estimates. Two types of uncertainty must be considered. First, aleatoric uncertainty is irreducible and caused by inherent noise in the data. Quantifying this uncertainty gives a lower bound on the SoH. Second, epistemic uncertainty is reducible and is caused by, among other factors, a lack of training data. Epistemic uncertainty can be used to identify if a test sample differs from the training samples, i.e., if it is Out-Of-Distribution (OOD).

In this paper, we estimate the SoH during discharge based on the current and voltage measurements obtained during charge. For this, we employ a Bidirectional Gated Recurrent Unit (Bi-GRU) neural network with attention. We estimate the aleatoric uncertainty using Simultaneous Quantile Regression (SQR), while we estimate the epistemic uncertainty by applying Orthonormal Certificates (OC). We test our approach on the fast charging dataset of Toyota. We achieve good results with a high accuracy, with a RMSE of only 0.00343 Ampere hours, and a good calibration. The model estimations become less accurate near the End of Life (EoL) of the batteries, but the corresponding data samples are correctly identified as OOD due to the high epistemic uncertainty.

1. Introduction

Batteries are a crucial in the transition towards a sustainable society, and are for instance employed in electric vehicles and in Unmanned Aerial Vehicles (UAVs). Battery Health Management is therefore becoming increasingly important (Thelen et al., 2024). In battery health management, a key quantity is the State of Health (SoH), which is the discharge capacity available in a battery after a full charge. This capacity decreases over time as the battery degrades. The SoH is estimated based on measurements obtained during charging, and is used to plan operations and battery replacements.

Physical battery models, such as equivalent circuit models (ECM) and electrochemical models (EM), are frequently used for battery SoH estimation. These models, however, require extensive prior knowledge about the battery (Zhang & Li, 2022), which is not always available in practice. In this paper, we therefore use a data-driven method instead.

Most data-driven methods for battery SoH estimation provide a single point estimate (Rasul, Abbas, Baek, & Kim, 2026). However, quantifying the uncertainty of the SoH estimate is crucial for employing SoH estimation models in practice, especially for safety-critical applications. First, uncertainty quantification enables the planning of operations based on the lower bound of the estimated SoH. In this way, the often negative consequences of overestimating the SoH, e.g., a crash of the UAV, can be avoided. Second, the quantified uncertainty indicates to the user when the SoH estimated by the model can, and more importantly cannot, be trusted. Specifically, the outcomes of a machine learning model are usually untrustworthy for Out-Of-Distribution (OOD) samples, i.e., for samples that differ significantly from (most) training samples. An uncertainty quantification method should thus yield a high uncertainty for OOD samples (Nemani et al., 2023).

In literature, various methods are employed for SoH estimation with uncertainty quantification. In (Roman, Saxena, Robu, Pecht, & Flynn, 2021), ensemble learning is used, while a Bayesian Neural Network (BNN) is employed in (Ke, Zhou, Zhu, & Peng, 2021) instead. Both ensemble learning and BNNs, however, are computationally expensive, which is undesirable for practical applications. In (Ke et al., 2021; Ke, Long, Yang, & Peng, 2024), the BNN is therefore approximated by applying Monte Carlo dropout. With MC dropout, however, the estimated uncertainty is highly dependent on the hyperparameters of the neural network, which makes these methods highly sensitive for user-defined inputs.

Conformal prediction and the quantile loss function are two alternative uncertainty quantification methods that require only little computational time and that are not dependent on any parameter or assumed distribution of the uncertainty. For instance, in (Rasul et al., 2026), ensemble learning is combined with conformal prediction, where the uncertainty of the SoH estimation is derived based on the predictions for a calibration set. In (Zhang, Zhang, Liu, Feng, & Xu, 2024), a quantile loss function is used to estimate the SoH with uncertainty. The main disadvantage of these methods, however, is that they only capture the aleatoric uncertainty.

The aleatoric uncertainty is the irreducible uncertainty that is inherent in the link between inputs and outputs of the model (Cabezas, Santos, Ramos, & Izbicki, 2025; Tagasovska & Lopez-Paz, 2019). The epistemic uncertainty, which is the reducible uncertainty caused by a lack of knowledge, such as a lack of data, is not captured by these methods. For OOD samples, there is a lack of similar training data, and the epistemic uncertainty is thus high. Both conformal prediction and quantile regression alone would therefore underestimate the uncertainty of OOD samples, erroneously indicating that the outcomes of a machine learning model can be trusted for these samples. In (Zhang et al., 2024), this is solved by adjusting their loss function based on the Wasserstein distance, but for this, they have to assume a distribution for the uncertainty.

In this work, we tackle this gap by estimating the SoH and quantify the uncertainty with two methods that are adopted from (Tagasovska & Lopez-Paz, 2019). First, we use Simultaneous Quantile Regression (SQR), with a special quantile loss function, to estimate the aleatoric uncertainty. Second, we use Orthonormal Certificates (OC) to estimate the epistemic uncertainty and detect OOD samples. Both methods require only little computational time, and are independent on user-defined inputs such as hyperparameters or an assumed distribution. Moreover, these methods have, to the best of our knowledge, not yet been applied for uncertainty quantification of the battery SoH. Last, by separately quantifying the aleatoric and epistemic uncertainty, we provide additional insights into the different sources of uncertainty.

As a SoH estimation model, we develop a Gated Recurrent Unit (GRU) neural network with attention. This neural network automatically extracts features from raw sensor data, thus eliminating the need for manual feature extraction as often used in battery SoH estimation (see for instance (Roman et al., 2021)). We test our approach on the Toyota battery fast charging dataset (Severson et al., 2019). The results show that a model trained with SQR provides narrow, but well calibrated, confidence intervals for the estimated SoH, while the model with OC successfully detects OOD samples.

Summarizing, the main contributions of this paper are:

- To the best of our knowledge, we are the first to apply SQR and OC to separately estimate the aleatoric and epistemic uncertainty, respectively, of the battery SoH estimation.
- We integrate these uncertainty quantification methods in a GRU neural network that automatically extracts feautres from raw sensor data, thus eliminating the manual feature extraction phase
- We test our results on the Toyota battery fast charging data set.

In the remainder of this paper, we first introduce our methodology in Section 2, followed by the case study and results in Section 3. We provide the conclusions in Section 4.

2. METHODOLOGY

In this Section, we first introduce the SoH estimation model in Section 2.1. We subsequently introduce the aleatoric uncertainty estimation method in Section 2.2, and the epistemic uncertainty estimation method in Section 2.3.

2.1. SoH estimation model

In this section, we introduce a SoH estimation model for batteries. First, let C denote the number of cycles a battery is used, where each cycle consists of first charging, and subsequently discharging a battery. We assume that we measure the voltage (V) and current (I) during charge. Let $V_c = [V_c^1, \ldots, V_c^{t(c)}]$ and $I_c = [I_c^1, \ldots, I_c^{t(c)}]$, $c \in \{1, \ldots, C\}$, denote the voltage and current measurements of cycle c during charge, respectively, with $t(c)$ the number of measurements during cycle c. We normalize the data of each charging cycle between -1 and $+1$ using min-max normalization. Our model can be extended to also include temperature measurements, if available.

We use V_c and I_c to estimate the capacity Q_c during discharge. The estimated SoH of cycle c then equals (Xiong, Li, & Tian, 2018):

$$\text{SoH}_c = \frac{Q_c}{Q_{\text{nom}}}, \tag{1}$$

where Q_{nom} denotes the initial, nominal capacity of a new battery.

The number of measurements per cycle c, $t(c)$, varies from cycle to cycle. We therefore estimate the SoH using a Recurrent Neural Network (RNN) with zero padding and masking. We specifically use a Bidirectional Gated Recurrent Unit (Bi-GRU) neural network. To enhance the explainability of the GRU and improve its ability to learn complex, long patterns, we add a hierarchical attention mechanism (Yang et al.,

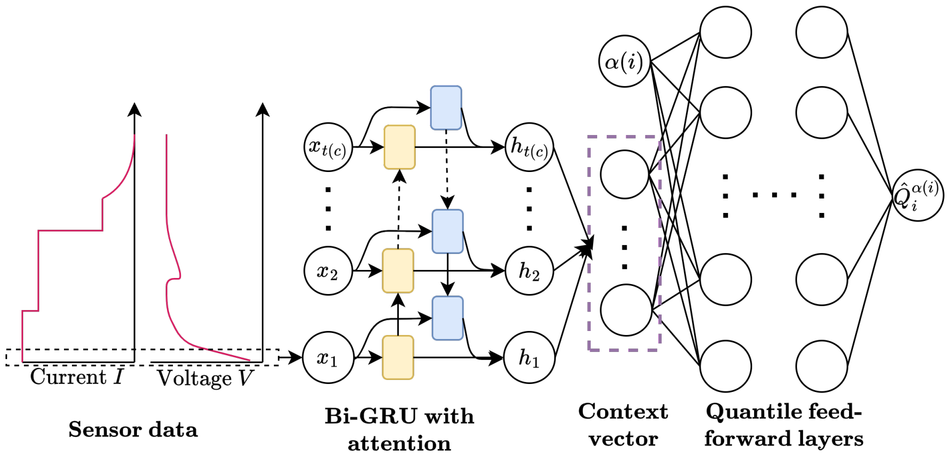

Figure 1. Overview of the considered SoH estimation model, where $x_i = [V_c^i, I_c^i]$, $i \in \{1, 2, \ldots, t(c)\}$ and h_t denotes the output vector of the BiGRU layer with hierarchical attention at time t.

2016). The output of the Bi-GRU attention layer with hierarchal attention is a context vector, which we subsequently input in several feedforward layers to predict the capacity Q_c. Figure 1 shows the model architecture.

2.2. Estimation of aleatoric uncertainty using SQR

The aleatoric uncertainty is the irreducible uncertainty that is inherent to the relationship between the inptus (voltage and current measurements) and output (capacity estimate) of the model. Aleatoric uncertainty can be estimated by training a model to minimize the quantile loss function (Koenker & Hallock, 2001). The quantile loss function for a considered quantile $\alpha \in [0, 1]$, also called the pinball loss function, is:

$$\mathcal{L}(\alpha) = \sum_{i=1}^{N} w_i^{\alpha} |Q_i - \hat{Q}_i^{\alpha}|, \qquad (2)$$

with

$$w_i^{\alpha} = \begin{cases} 1-\alpha & Q_i - \hat{Q}_i^{\alpha} < 0 \\ \alpha & Q_i - \hat{Q}_i^{\alpha} \geq 0 \end{cases}. \qquad (3)$$

Here, Q_i is the true capacity of sample i, $\hat{Q}_i^{\alpha}$ is the output of the model for sample i and quantile α, and N is the number of samples. The quantile loss function is based on the standard $L1$ loss function, and the optimal loss is reached by predicting $\hat{Q}_i^{\alpha}$ such that it equals the α-quantile of the capacity of sample i (Tagasovska & Lopez-Paz, 2019). For instance, for the $\alpha = 0.1$ quantile estimate of the capacity $\hat{Q}_i^{0.1}$, the loss is minimized if we predict $\hat{Q}_i^{0.1}$ such that there is a 10% probability that the true capacity falls below $\hat{Q}_i^{0.1}$, and that there thus is a 90% probability that the true capacity exceeds $\hat{Q}_i^{0.1}$.

In standard quantile regression, a separate model is trained for each quantile α. One can subsequently estimate the Cumulative Distribution Function (CDF) of the capacity of a single sample i by computing the α quantile estimate for different values of α. Unfortunately, this might lead to "quantile crossing", which means that the estimated quantiles do not monotonically increase (Waltrup, Otto-Sobotka, Kneib, & Kauermann, 2015). For instance, we get an invalid CDF if we estimate that the $\alpha = 0.2$ quantile equals 1.7 Ah, while the $\alpha = 0.3$ quantile estimate is only 1.5 Ah. Moreover, training a separate model for each quantile is computationally expensive.

We therefore instead use Simultaneous Quantile Regression (SQR) (Tagasovska & Lopez-Paz, 2019). When training the model with SQR, we sample a random quantile $\alpha(i)$ from the uniform distribution for each data sample i, i.e., $\alpha(i) \sim \mathcal{U}[0, 1]$. The loss function thus becomes:

$$\mathcal{L} = \sum_{i=1}^{N} w_i^{\alpha(i)} |Q_i - \hat{Q}_i^{\alpha(i)}|, \qquad (4)$$

with $w_i^{\alpha(i)}$ as defined in Eq. (3). For each sample i, $\alpha(i)$ is added as input to the first feedforward layer of the neural network (see Figure 1). When training our model for multiple epochs, we resample $\alpha(i)$ for each sample i in each epoch. By training our model with the SQR loss function, we have a single model with which we can predict the quantile $\hat{Q}_i^{\alpha}$ of any sample i for any value of $\alpha \in [0, 1]$.

2.3. Estimation of epistemic uncertainty

The epistemic uncertainty is the reducible uncertainty that is caused by a lack of knowledge, such as a lack of representative training data. We estimate the epistemic uncertainty using Orthonormal Certificates (OC) (Tagasovska & Lopez-Paz, 2019). First, we train the SoH estimation model with the SQR loss function (Eq. (4)). Let $g_{i,\alpha}$ denote the vector with the output of the penultimate layer of the trained model, when we input sample i and a quantile $\alpha \in [0, 1]$. We subsequently remove the last layer of the neural network, and instead add a new OC layer with weights w^{OC}. The input of this OC layer is $g_{i,\alpha}$, and the output is a vector of dimension m, denoted by $y_{i,\alpha}^{\text{OC}}$. Here, m denotes the certificate size, which is a tunable hyperparameter. The OC layer has no activation function or bias. This is illustrated in Figure 2.

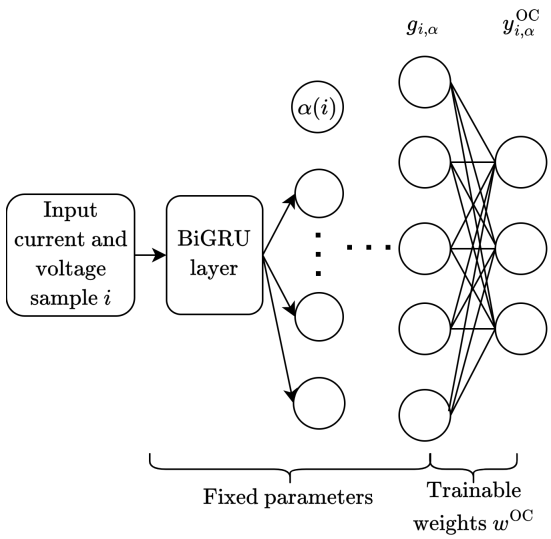

Figure 2. The OC layer used to estimate the epistemic uncertainty. In this example, $m = 3$.

We subsequently retrain the neural network, but we only update the weights w^{OC} of the new OC layer. All other parameters of the neural network are frozen. We minimize the following loss function:

$$\mathcal{L} = \frac{1}{N} \sum_{i=1}^{N} \mathcal{L}' \left(0, y_{i,\alpha}^{\text{OC}}\right) + \frac{\lambda}{m} \left\| w^{\text{OC}} \left(w^{\text{OC}}\right)^T - I_m \right\|^2 \quad (5)$$

Here, $\mathcal{L}'$ equals:

$$\mathcal{L}' \left(0, y_{i,\alpha}^{\text{OC}}\right) = \sum_{j=1}^{m} w_{i,j}^{\alpha(i)} \left|0 - y_{i,\alpha}^{\text{OC}}[j]\right|, \quad (6)$$

with

$$w_{i,j}^{\alpha(i)} = \begin{cases} 1 - \alpha(i) & 0 - y_{i,\alpha}^{\text{OC}}[j] < 0 \\ \alpha(i) & 0 - y_{i,\alpha}^{\text{OC}}[j] \geq 0 \end{cases}, \quad (7)$$

where $\alpha(i)$ is sampled from an uniform distribution $\mathcal{U}[0, 1]$ and $y_{i,\alpha}^{\text{OC}}[j]$ denotes the j^{th} element of the vector $y_{i,\alpha}^{\text{OC}}$. Note that $\mathcal{L}'$ equals the loss function in Eq. (4), except that the ground truth is zero for each element in $y_{i,\alpha}^{\text{OC}}$. We thus train the neural network to map each element in $y_{i,\alpha}^{\text{OC}}$ to zero.

The second term in Eq. (6) prevents that the calculation of different elements in $y_{i,\alpha}^{\text{OC}}$ is (nearly) the same, i.e., to prevent that the weights w^{OC} are equal for each output $y_{i,\alpha}^{\text{OC}}[j]$, $j = 1, \ldots, m$. Here, λ is a hyperparameter, I_m is the $m \times m$ identity matrix, and we employ the Frobenius norm:

$$\left\| w^{\text{OC}} \left(w^{\text{OC}}\right)^T - I_m \right\|^2 = \sum_{i=1}^{m} \sum_{j=1}^{m} \left(w^{\text{OC}} \left(w^{\text{OC}}\right)^T - I_m \right)_{ij}^2 \quad (8)$$

where A_{ij} denotes the element in the i^{th} row and j^{th} column of a matrix A.

After retraining, the model learned to map the output of the penultimate layer, $g_{i,\alpha}$, to a vector with zeroes for the training samples. However, if a test sample differs from a training sample, i.e., if it is OOD, we expect that the model does not succeed in mapping it to a vector with zeroes. We therefore compute the epistemic uncertainty u_i of a sample i by taking the mean of $y_{i,0.5}^{\text{OC}}$.

$$u_i = \frac{1}{m} \sum_{j=1}^{m} \left(y_{i,0.5}^{\text{OC}}[j]\right)^2. \quad (9)$$

Here, we enforce $\alpha = 0.5$ to prevent that sampling different values of α influences the epistemic uncertainty estimate.

The epistemic uncertainty is unitless, and can therefore not directly be integrated in the CDF of the capacity, as estimated with SQR. We instead use the estimated epistemic uncertainty to label samples either "in distribution", if u_i is close to zero, or OOD, if u_i is significantly higher than zero. Specifically, as in (Tagasovska & Lopez-Paz, 2019), we assume that most training data samples are in distribution. We then calculate the 95^{th} percentile of the epistemic uncertainty across all training samples as an OOD threshold. When validating and testing the model, we label a sample as OOD if the estimated epistemic uncertainty is higher than this threshold.

3. CASE STUDY AND RESULTS

In this Section, we first introduce the case study in Section 3.1. We subsequently introduce the evaluation metrics in Section 3.2. Last, we present the SoH estimation results with aleatoric uncertainty in Section 3.3, while we present the epistemic uncertainty in Section 3.4.

3.1. Case study

We apply our methodology to the fast charging dataset published by the Toyota Research Institute (Severson et al., 2019). We consider batch 3 of this dataset, containing 46 $1.1Ah$ phosphate based lithium-ion batteries. During a cycle, a battery is charged with a battery-specific 2-phase fast charging protocol, and subsequently discharged with the traditional Constant Current - Constant Voltage (CC-CV) procedure (dos Reis, Strange, Yadav, & Li, 2021). When charging the battery, there are two "blocks" of Constant Current (CC), where the current is significantly higher than in the traditional CC-CV charging procedure. When the battery is charged for roughly 80%, the standard CC-CV procedure is followed instead. These cycles continue until the battery reaches its End-of-Life (EoL), in a controlled environment with a constant temperature of 30°.

Table 1. The considered training, validation and test split.

Set	Battery names ("B3_...")
Train	1, 2, 3, 4, 6, 8, 9, 12, 14, 15, 16, 17, 18, 19, 20, 21, 22, 25, 27, 31, 33, 44, 35, 36, 37, 38, 41, 42, 43, 44, 45, 46
Validation	7, 10, 13, 24, 28, 39, 40
Test	5, 11, 23, 26, 29, 30, 32

We randomly divide the 46 batteries with a 70% training, 15% testing and 15% validation split, as given in Table 1. With this, we finetune the hyperparameters of the SoH estimation model, as given in Table 2. We train the model in PyTorch using the AdamW optimizer.

Table 2. Hyperparameters for the SoH estimation model.

Hyperparamer	Value
BiGru with attention	
Hidden state size	16
Attention size	32
Attention dropout	0.1
Feedforward layers	
Number of layers	4
Number of neurons per layer	33/128/128/1
Activation function	ReLU
Data	
Sampling rate	10 sec.
Optimizer	
Number of epochs	150
Weight decay	1.0E-05
Mini-batch size	64
Initial learning rate l_r	1.0E-03
Reduce l_r factor	0.2
Reduce l_r patience	5 epochs
Reduce l_r sensitivity	1.0E-05
Early stop patience	10 epochs
Early stop sensitivity	1.0E-06

3.2. Evaluation metrics

We evaluate the median predictions of our model with $\alpha = 0.5$ using the RMSE and MAE:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(Q_i - \hat{Q}_i^{0.5})^2} \tag{10}$$

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N}|Q_i - \hat{Q}_i^{0.5}|, \tag{11}$$

where N denotes the number of test samples.

We also evaluate the predicted CDF of the capacity, as estimated with the aleatoric uncertainty, with a calibration curve (Nemani et al., 2023; De Pater & Mitici, 2022). For instance, assume that we consider a quantile of $\alpha = 0.3$, and that we make for $N = 1000$ test samples, a prediction of the capacity $Q_i^{0.3}$ ($i \in \{1, 2, \ldots, 1000\}$). For one individual sample i, we then thus predict that the capacity is lower than $Q_i^{0.3}$ with a 30% probability, and that the capacity is higher than $Q_i^{0.3}$ with 70% probability. Ideally, if we would do this for a 1000 samples, we would thus observe in reality that for 300 samples, the true capacity is lower than $Q_i^{0.3}$, and that for 700 samples, the true capacity is higher than $Q_i^{0.3}$. A calibration curve is used to test this desired property for different quantiles.

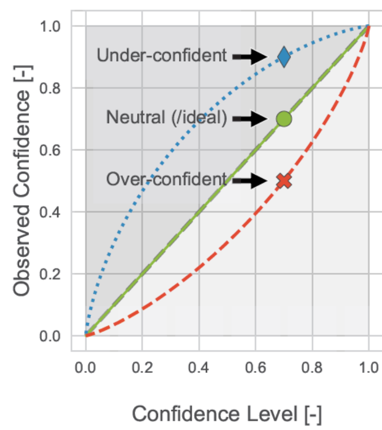

Figure 3. Overview of the ideal, an under-confident and a overconfident calibration curve.

Specifically, we define the observed confidence $\hat{c}^{\alpha}$ as:

$$\hat{c}^{\alpha} = \frac{1}{N}\sum_{i=1}^{N}\delta\left(Q_i \in [\hat{Q}_i^{\alpha}, +\infty)\right), \tag{12}$$

with

$$\delta\left(Q_i \in [\hat{Q}_i^{\alpha}, +\infty)\right) = \begin{cases} 1 & Q_i \in [\hat{Q}_i^{\alpha}, +\infty) \\ 0 & \text{Otherwise} \end{cases} \tag{13}$$

We aim to observe a confidence level of $\hat{c}^{\alpha} = 1 - \alpha$. We are overconfident if we observe that $\hat{c}^{\alpha} < 1 - \alpha$. For instance, in our example with $\alpha = 0.3$, we aim to observe a confidence level of $\hat{c}^{0.3} = 0.7$. If we observe a smaller confidence level, for instance $\hat{c}^{0.3} = 0.6$, then we predict that the probability that the true capacity is larger than $Q_i^{0.3}$ is 70%, but we observe that it is only 60%. The probability that the true capacity is lower than $Q_i^{0.3}$ is then 40% instead of 30%. Such overconfident predictions might endanger the safety of operations in safety-critical battery applications. If we observe that $\hat{c}^{\alpha} > 1 - \alpha$, we are underconfident, which might lead to underusing the capabilities of the battery. This is illustrated in Figure 3.

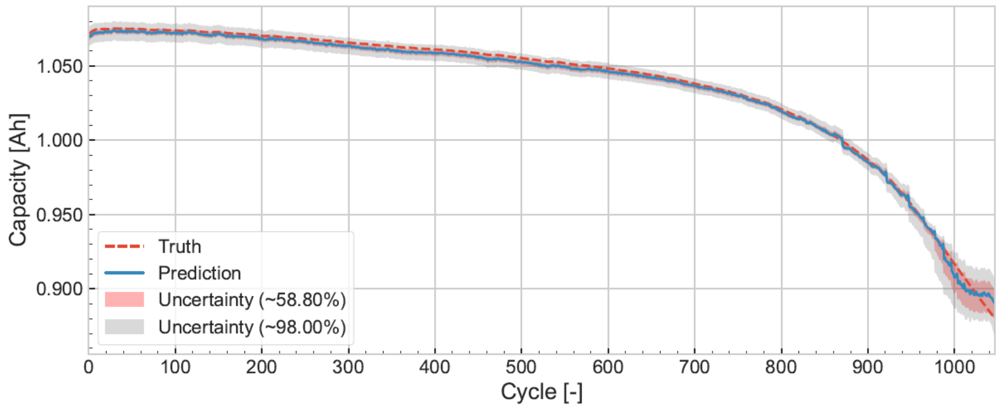

Figure 4. Estimated capacity, with aleatoric uncertainty, for battery 5.

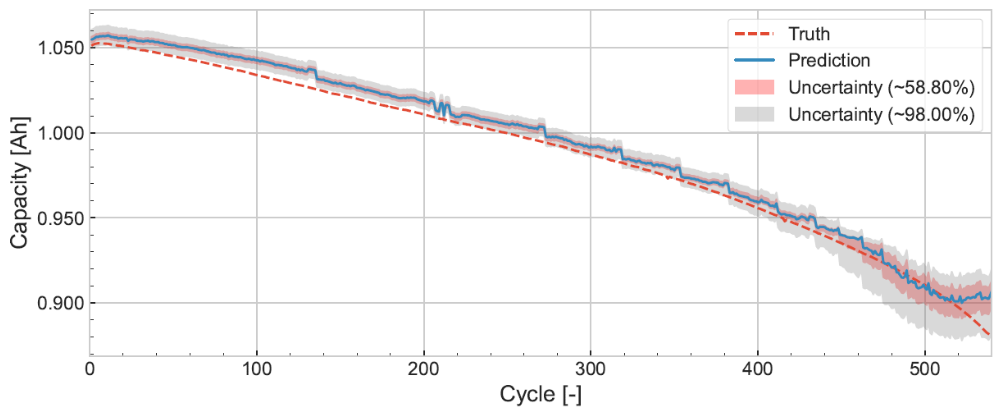

Figure 5. Estimated capacity, with aleatoric uncertainty, for battery 29.

3.3. Results: Estimated capacity and the aleatoric uncertainty

In this section, we evaluate the estimated capacity and the aleatoric uncertainty of the batteries. For the median predictions, with $\alpha = 0.5$, we obtain a RMSE of 0.00343 Ah, and a MAE of 0.00231 Ah. Our model is thus able to accurately predict the capacity with only small errors. Figure 4 and 5 further illustrate the results for battery 5 and 29, respectively. For battery 5, the estimated capacity is highly accurate, with a very narrow confidence interval that contains the true capacity. For battery 29, however, the capacity is consistently slightly overestimated. The 98% confidence interval of the estimated capacity for this battery is also very narrow, but unfortunately, the true capacity does not always fall inside.

Figure 4 and 5 also show that near the EoL, the median estimated capacity is higher than the true capacity, Moreover, the estimations are more uncertain, as indicated by the wider confidence intervals. This is the case for all test batteries. This might be because the charging time usually decreases as the battery ages, due to the decreased capacity. Since we use a variable length sequence as input to the model, the model

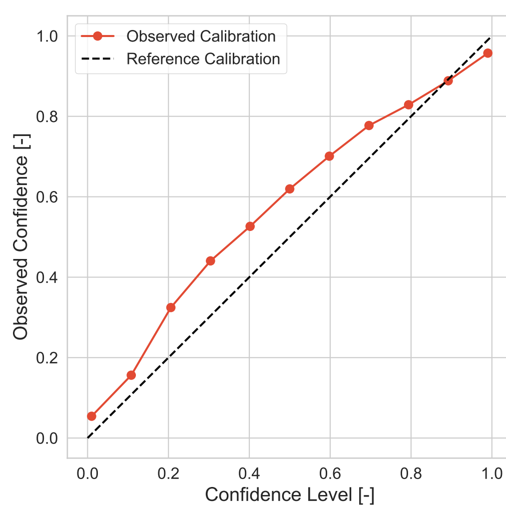

Figure 6. Calibration curve for all test batteries.

may have learned that the capacity decreases when the input sequence becomes shorter. For the last few cycles before the EoL, however, the charging time instead increases again.

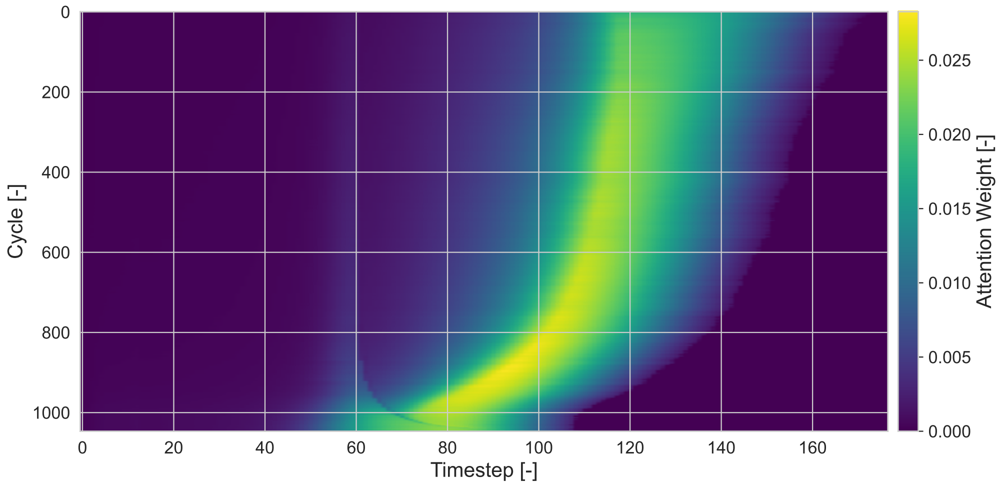

Figure 7. The attention weights of the model, as a function of time, for each cycle of battery 5

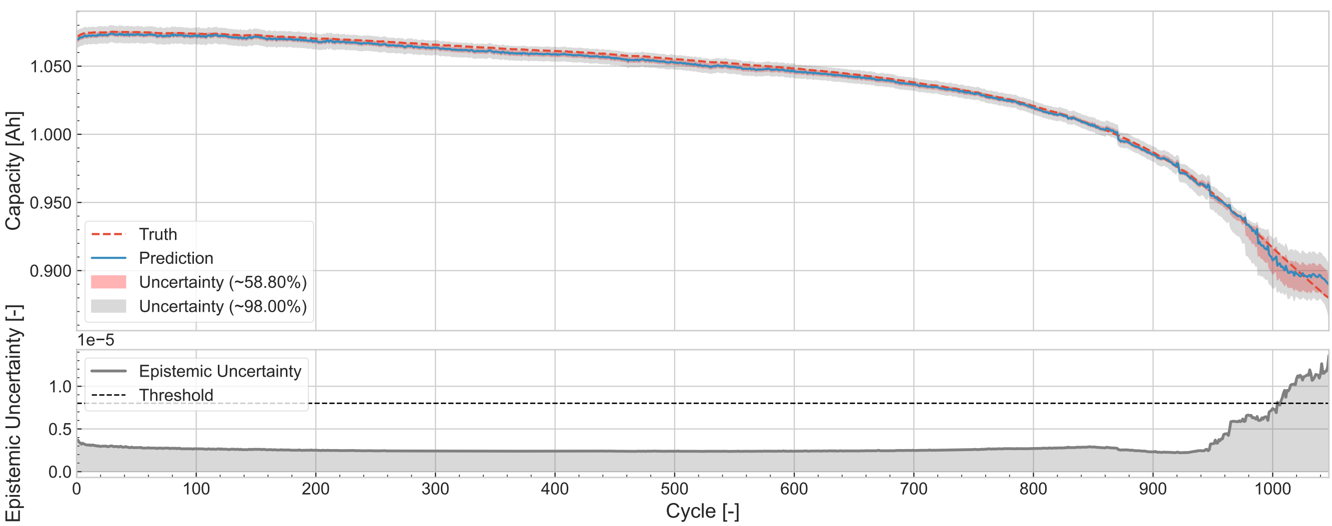

Figure 8. The estimated epistemic uncertainty for battery 5.

This is because during the 2-phase fast charging procedure, the voltage threshold is reached relatively early. The current is therefore gradually decreased, which increases the charging time.

Figure 6 shows the calibration curve. We observe that we achieve a near-ideal calibration, where we are slightly under-confident for most confidence levels. Moreover, in Figure 4 and 5, we observe that the confidence intervals are very narrow. This was observed for all test batteries. We can thus estimate the capacity accurately with a high confidence. However, when considering a high confidence level of 99%, the model becomes overconfident: We predict that the probability that the capacity is lower than $Q_i^{0.99}$ is 1%, while in reality, it is larger than 1%. This is undesirable, as for safety-critical applications, operations and battery replacements will be planned based on these estimates at high confidence levels.

Last, we visualize the attention weights in Figure 7 for battery 5 over the different cycles. The model usually focuses its attention on the measurements at the end of the charging cycle, in the traditional CC-CV part. As the charging time decreases when the battery degrades (except near the EoL), the attention weights shift slightly backwards in time as well. This is in line with the research in (Roman et al., 2021), where features for SoH estimation are created based solely on the CC-CV part. However, near the EoL, the model also focuses on the fast charging part. This may be an additional reason for the relatively poor model performance near the EoL.

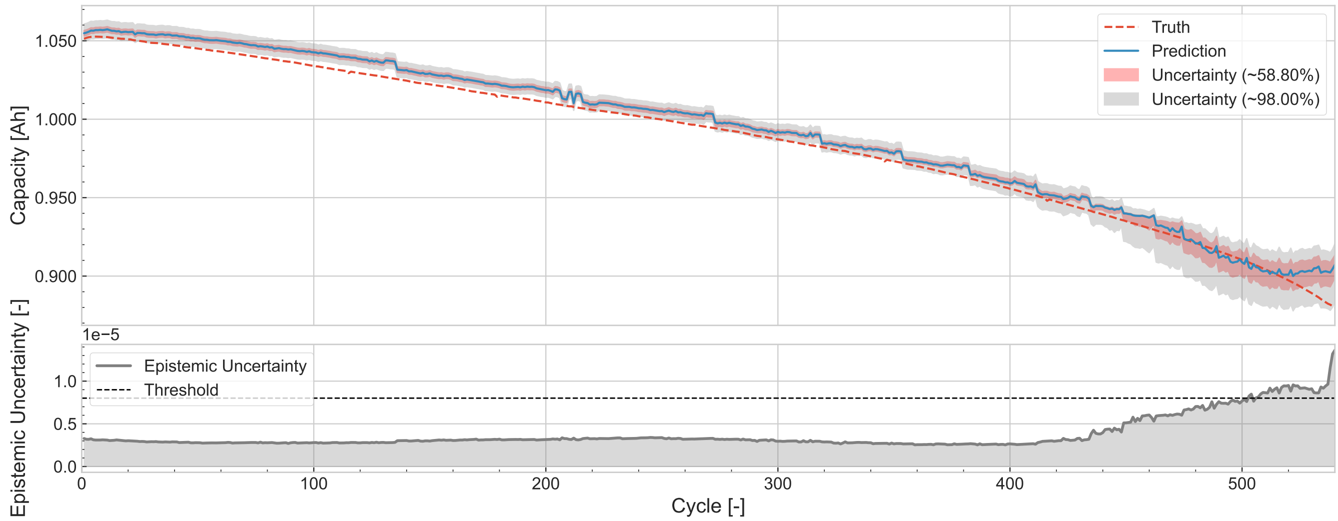

Figure 9. The estimated epistemic uncertainty for battery 29.

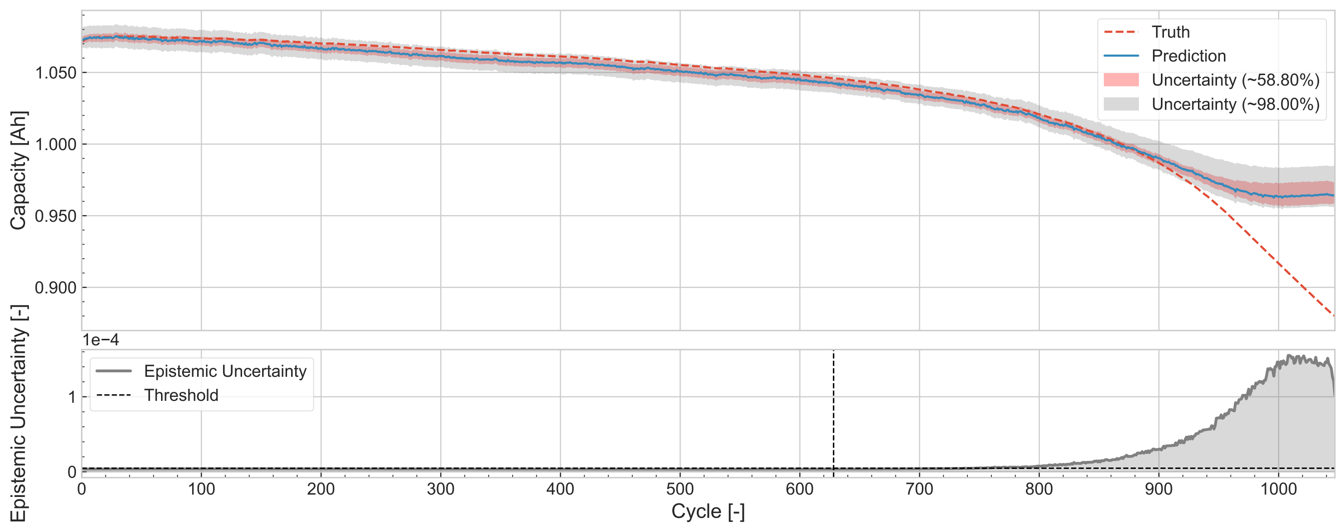

Figure 10. The estimated epistemic uncertainty for battery 5, when masking the last 40% of the cycles of each training battery during training.

3.4. Results: Epistemic uncertainty

We estimate the epistemic uncertainty using an OC layer with 128 neurons, which we trained using the Adam optimizer for 10 epochs, a regularization factor λ of 1 and a learning rate of 1.0E-3. The OOD threshold equals $8.009{\cdot}10^{-6}$, i.e., we label a sample as OOD if its epistemic uncertainty exceeds this threshold. Figure 8 shows the results for battery 5, while Figure 9 shows the results of battery 29. The epistemic uncertainty is consistently low for the first 90% of the cycles of both batteries. In general, for all test batteries, no OOD samples are detected in the first 90% of the cycles. This is in line with the accurate estimates of the capacity for the test batteries, except near the EoL. The epistemic uncertainty thus correctly indicates that the model can be trusted in this region.

Near the EoL of battery 5 and 29, however, the epistemic uncertainty increases. This also occurs for the other test batteries: In the last 10% of the cycles, on average 36% of the samples is labeled as OOD. This is in line with the relatively inaccurate capacity estimations near the EoL (see Section 3.3). This shows that by splitting up the estimation of the aleatoric and the epistemic uncertainty, we can correctly identify when the SoH estimation model cannot be trusted: In this regions, the epistemic uncertainty is high and we have many OOD samples. In practice, this would allow human intervention, for instance by manually inspecting the sensor data, or by taking a larger safety buffer when planning operations.

Interestingly, only 1.5% of the training samples are near the EoL and have an increased charging time compared to the previous cycles. Test samples near the EoL with a relatively long charging time are thus indeed different from the majority of the training samples. This shows that the epistemic uncertainty correctly labels test samples as OOD. However, for reliable SoH estimations near the EoL of the battery, we would need to either employ a different model with a constant input length, or to consider another dataset with more samples near the EoL of the battery.

We subsequently train our model with only the first 60% of the cycles per training battery, while we estimate the capacity of all cycles of the test batteries. In a practical application, such a situation would occur if an operator decides to use the batteries for a longer period of time. Figure 10 shows the results for battery 5. After 60% of the life cycles, the model still provides accurate capacity estimations, even though the model was not trained on samples of this region. The epistemic uncertainty remains relatively low as well, indicating that these predictions are indeed relatively trustworthy. After roughly 900 cycles, however, the capacity estimation becomes less accurate. This is not captured by the aleatoric uncertainty, as the confidence intervals remain narrow. The epistemic uncertainty, however, clearly increases, thus correctly indicating that the model outcomes are not trustworthy anymore.

4. CONCLUSION

In this paper, we use a Bi-GRU neural network with hierarchical attention to estimate the capacity of a battery based on the voltage and current measurements obtained during charge. We use SQR to estimate the aleatoric uncertainty of these estimates, while we employ OCs to estimate the epistemic uncertainty. We test our approach on the fast charging dataset published by the Toyota Research Institute (Severson et al., 2019). Our model provides accurate estimates, with a RMSE of only 0.00323 Ah. The estimations are especially accurate during the first 90% of the cycles of each test battery. In this region, the aleatoric uncertainty provides a narrow confidence interval for the capacity, but remains well calibrated, though slightly underconfidendent. Near the EoL, however, the capacity estimations become less accurate. This is reflected by the epistemic uncertainty, which increases near the EoL, thus correctly indicating that the model becomes less trustworthy.

For future research, we will extend our model by predicting the RUL of batteries by estimating how the SoH will decrease over time. Moreover, we will test our epistemic uncertainty approach with other experiments. Last, we will compare our approach to different benchmarks, varying both the uncertainty quantification method and the SoH estimation model. Most importantly, we will also develop a model that has a fixed sequence length as input. This might improve the predictions near the EoL and make the model more robust to different hyperparameters.

REFERENCES

Cabezas, L., Santos, V. S., Ramos, T. R., & Izbicki, R. (2025). Epistemic uncertainty in conformal scores: A unified approach. *arXiv preprint arXiv:2502.06995*.

De Pater, I., & Mitici, M. (2022). Novel metrics to evaluate probabilistic remaining useful life prognostics with applications to turbofan engines. *PHM Society European Conference*, *7*, 96–109.

dos Reis, G., Strange, C., Yadav, M., & Li, S. (2021). Lithium-ion battery data and where to find it. *Energy and AI*, *5*, 100081.

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.

Ke, Y., Zhou, R., Zhu, R., & Peng, W. (2021). State of Health estimation of lithium-ion battery with uncertainty quantification based on Bayesian deep learning. In *2021 3rd international conference on system reliability and safety engineering (srse)* (p. 12-18).

Koenker, R., & Hallock, K. F. (2001). Quantile regression. *Journal of Economic Perspectives*, *15*, 143–156.

Nemani, V., Biggio, L., Huan, X., Hu, Z., Fink, O., Tran, A., … Hu, C. (2023). Uncertainty quantification in machine learning for engineering design and health prognostics: A tutorial. *Mechanical Systems and Signal Processing*, *205*, 110796.

Rasul, M. J., Abbas, A., Baek, J., & Kim, J. (2026). A hybrid ensemble learning framework with uncertainty quantification for state-of-health estimation in lithium-ion batteries. *Measurement*, 120528.

Roman, D., Saxena, S., Robu, V., Pecht, M., & Flynn, D. (2021). Machine learning pipeline for battery state-of-health estimation. *Nature Machine Intelligence*, *3*, 447-456.

Severson, K. A., Attia, P. M., Jin, N., Perkins, N., Jiang, B., Yang, Z., … Braatz, R. D. (2019). Data-driven prediction of battery cycle life before capacity degradation. *Nature Energy*, *4*, 383–391.

Tagasovska, N., & Lopez-Paz, D. (2019). Single-model uncertainties for deep learning. *Advances in neural information processing systems*, *32*.

Thelen, A., Huan, X., Paulson, N., Onori, S., Hu, Z., & Hu, C. (2024). Probabilistic machine learning for battery health diagnostics and prognostics—review and perspectives. *NPJ Materials Sustainability*, *2*, 14.

Waltrup, L., Otto-Sobotka, F., Kneib, T., & Kauermann, G. (2015). Expectile and quantile regression—David and Goliath? *Statistical Modelling*, *15*, 433-456.

Xiong, R., Li, L., & Tian, J. (2018). Towards a smarter battery management system: A critical review on battery state of health monitoring methods. *Journal of Power Sources*, *405*, 18–29.

Yang, Z., Yang, D., Dyer, C., He, X., Smola, A., & Hovy, E. (2016). Hierarchical attention networks for document classification. In K. Knight, A. Nenkova, & O. Rambow (Eds.), *Proceedings of the 2016 conference of the north American chapter of the association for computational linguistics: Human language technologies* (pp. 1480–1489). San Diego, California: Association for Computational Linguistics. doi: 10.18653/v1/N16-1174

Zhang, Y., & Li, Y.-F. (2022). Prognostics and health management of lithium-ion battery using deep learning methods: A review. *Renewable and sustainable energy reviews*, *161*, 112282.

Zhang, Y., Zhang, M., Liu, C., Feng, Z., & Xu, Y. (2024). Reliability enhancement of State of Health assessment model of lithium-ion battery considering the uncertainty with quantile distribution of deep features. *Reliability Engineering & System Safety*, *245*, 110002.