Array 24 (2024) 100367

Contents lists available at ScienceDirect

Array

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

Dual-model approach for one-shot lithium-ion battery state of health sequence prediction

Slimane Arbaoui [a,*], Ahmed Samet [a], Ali Ayadi [a], Tedjani Mesbahi [b], Romuald Boné [a]

[a] SDC team, ICube laboratory, INSA Strasbourg, University of Strasbourg, 24 Bd de la Victoire, Strasbourg, 67000, France
[b] SMH team, ICube laboratory, INSA Strasbourg, University of Strasbourg, 24 Bd de la Victoire, Strasbourg, 67000, France

ARTICLE INFO

Keywords:
Deep learning
Battery
Explainability
SOH estimation
End of life
Transformers
LSTM

ABSTRACT

Lithium-ion batteries play a crucial role in powering various applications, including Electric Vehicles (EVs), underscoring the importance of accurately estimating their State Of Health (SOH) throughout their operational lifespan. This paper introduces two novel models: a Transformer (TOPS-SoH) and a Long Short-Term Memory based (LSTM-OSoH) for One-shot Prediction of SOH. The LSTM-OSoHexcels in accuracy, achieving a Masked Mean Absolute Error (MMAE) of less than 0.01 for precise SOH estimation, while the TOPS-SoHdemonstrates simplicity and efficiency, with accuracy comparable to state-of-the-art models. The TOPS-SoHmodel also offers additional interpretability by providing insights into the attention scores between inputs and outputs, highlighting the cycles used for estimation. These models were trained using the MIT battery dataset, with auto-encoders employed to reduce the dimensionality of the input data. Additionally, the models' effectiveness was validated against a Bidirectional LSTM (*BiLSTM*) baseline, demonstrating superior performance in terms of lower MMAE, MMSE, and MAPE values, making them highly suitable for integration into Battery Management Systems (BMS). These findings contribute to advancing SOH estimation up to the End Of Life (EOL), which is crucial for ensuring the reliability and longevity of lithium-ion batteries in diverse applications.

1. Introduction

In the realm of energy storage, lithium-ion batteries serve as the cornerstone technology powering a myriad of applications, from portable electronics to Electric Vehicles (EVs). Central to ensuring the longevity and reliability of these batteries is the ability to accurately estimate their State Of Health (SOH) throughout their operational lifespan. This metric compares the current capacity of the battery with its original capacity when new and provides the amount of capacity retained in a scale of 0 to 100. Around 80% SOH, most batteries reach a turning point in which they may no longer meet the performance, safety and range requirements expected for an EV, leading manufacturers and users to consider it unsuitable for continued use in that application [1]. Advancing precise techniques to assess the SOH of batteries remains a compelling focus of research due to the formidable challenges posed by their intricate electrochemical dynamics. Existing commercial sensors fall short in directly measuring battery SOH, necessitating reliance on inferred estimations from parameters like charging/discharging currents, voltages, sampling intervals, temperatures, and more [2].

Numerous research endeavours have proposed machine learning models to tackle the task of SOH estimation in batteries, yielding promising results despite the inherent black-box nature of many of these models and the complexity that make them expensive in terms of storing and running, especially in an embedded system. Accurate estimation of SOH throughout a battery's lifecycle is imperative for two primary reasons. Firstly, it facilitates proactive maintenance strategies, enabling timely interventions or replacements to mitigate performance degradation risks and maintain operational safety. Secondly, precise SOH estimation empowers optimization of battery utilization, leading to extended lifespan and a reduction in costs [3]. Battery Management Systems (BMS) play a crucial role in achieving these objectives by accurately monitoring and controlling battery operations to ensure safety and optimal performance. A BMS is an electronic system that manages rechargeable batteries by protecting them from operating outside safe limits, monitoring their state, calculating secondary data like SOH, reporting that data, controlling the battery environment, authenticating it, and balancing the cells [4]. By providing precise SOH estimations, BMS enables proactive maintenance schedules based on the battery's health condition, mitigating risks and ensuring operational safety.

* Corresponding author.
E-mail addresses: slimane.arbaoui@insa-strasbourg.fr (S. Arbaoui), ahmed.samet@insa-strasbourg.fr (A. Samet), ali.ayadi@insa-strasbourg.fr (A. Ayadi), tedjani.mesbahi@insa-strasbourg.fr (T. Mesbahi), romuald.bone@insa-strasbourg.fr (R. Boné).

https://doi.org/10.1016/j.array.2024.100367
Received 9 June 2024; Received in revised form 25 September 2024; Accepted 29 September 2024
Available online 3 October 2024

However, a notable gap exists in the literature concerning models capable of estimating the entire SOH curve across its entire lifespan. Only a few models undertake this comprehensive task [5], and even fewer leverage transformers models for such estimation, which have gained prominence in sequence-to-sequence estimation tasks, renowned for their parallelizable nature and lightweight architecture. This paper aims to fill this gap by presenting two novel models: a Transformer (TOPS-SoH) and a Long Short-Term Memory based (LSTM-OSoH) for One-shot Prediction of SOH until reaching the first End Of Life (EOL) at 80% of battery's health. Each model has a strength: the LSTM-OSoHoffers perfect accuracy for precise SOH estimation, but may be expensive in terms of storage and energy. On the other hand, the TOPS-SoHmodel is simpler, has a smaller storage size, and is suitable for integration into a BMS. Its reduced number of parameters also aids in energy efficiency. Moreover, the model provides transparency by allowing the examination of attention between inputs and outputs, offering insight into the input cycles that influence the model's predictions, a feature often overlooked by most approaches, which can limit their applicability in the automated vehicle sector [6]. The main contributions of this paper can be summarized as follows:

- Data encoding: We initiated our approach by building an encoder to reduce the dimensionality of the data while preserving essential information.
- Initial model development: We trained a baseline model using a simple LSTM configuration to establish a foundational comparison.
- Transformer model design: We developed two Transformer models, each differing in their method of calculating attention scores, to evaluate their performance and efficiency.
- Comparison with existing solutions: We benchmarked our models against the only one-shot prediction model available in the literature, utilizing the code from the GitHub repository https://git.rwth-aachen.de/isea/battery-degradation-trajectory-prediction provided in the referenced paper.

The remainder of this paper is structured as follows: Section 2 provides an overview of related work in the field of SOH estimation. Section 3 introduces the main concepts of the neural network models used in this work. Section 4 details the proposed approach, outlining its methodology and steps. Section 5 presents a comprehensive analysis of the results obtained through our approach. Section 6 concludes the paper based on the findings and implications discussed in the paper.

2. State of the art

Current SOH estimation methods are divided into two main types: model-based and data-driven methods [7]. Model-based methods rely on mathematical models that represent the underlying physics and electrochemical processes within the battery, such as Equivalent Circuit Models (ECMs). ECMs are widely used for estimating the SOH of batteries, representing their electrochemical behavior using electrical components like resistors, capacitors, and voltage sources [8]. The simplest ECM represents the battery as an ideal voltage source in series with an internal resistance, known as the Rint model [9]. This model accounts for the voltage drop due to the internal resistance, which varies with the battery's state of charge (SOC), SOH, and temperature [9]. To improve the model's accuracy, researchers have introduced additional components to capture the battery's nonlinear behavior. One such enhancement is the RC model, which includes a surface capacitor and a bulk capacitor to represent the battery's charge storage capabilities, along with three resistors to account for different polarization effects [9]. However, ECMs also have limitations. Simple ECMs may have poor estimation accuracy, while complex models with many parameters can be computationally expensive. Accurately identifying the ECM parameters and relating them to SOH remains challenging, especially under dynamic operating conditions. In contrast, data-driven methods leverage machine learning algorithms to analyze large datasets of battery performance metrics and environmental factors [10]. These methods do not rely on explicit models of battery behavior but instead learn patterns and relationships directly from the data. By training on historical data from a diverse range of batteries, data-driven approaches can potentially capture complex interactions that are difficult to model analytically. However, data-driven methods may struggle to generalize to new operating conditions or battery chemistries that differ significantly from the training data. Additionally, they may require substantial computational resources for training and deployment.

In this paper we focus on data-driven methods highlighting, noteworthy techniques in this area including the hybrid model proposed by Gu et al. [11], which integrates Convolutional Neural Network (CNN) and Transformer architectures to predict SOH values. They utilized Principal Component Analysis (PCA) to distill three features from an initial set of four (temperature, voltage, current, and capacity), demonstrating remarkable accuracy on the NASA dataset with Root Mean Square Error (RMSE) less than 0.55%. Moreover, Fan et al. [12] introduced a T-LSTM model, combining Transformer and LSTM layers for SOH estimation based on historical voltage data, resulting in a notable reduction in error compared to simpler LSTM models. Zhang et al. [13] introduce a method using LSTM neural networks to estimate SOH. It incorporates health features from charging curves and Incremental Capacity Analysis (ICA) and optimizes hyperparameters with the Adam algorithm [14] for improved accuracy. Experimental validation across batteries with varying capacities confirms its effectiveness in estimating capacity degradation accurately with a Mean Absolute Error (MAE) of 4.6%. Additionally, it analyzes the impact of different training set lengths on capacity estimation, showing consistently high accuracy even with smaller sets.

Another method, employing a Radial Basis Function (RBF) neural network, was proposed by Chang et al. [15]. They used three hardware components: a battery voltage detection interface, discharge current detection interface, and battery impedance measurement interface, all integrated with a microprocessor for digital control. The RBF neural network, serving as the software of the SOH estimator, consists of an input layer, a hidden layer, and an output layer. To validate the accuracy of the method, experiments were conducted under various discharging conditions using LiFePO4 (LFP) batteries. The accuracy of the SOH estimation method was assessed using two metrics: the Maximum Absolute Percentage Error (MaxAPE) and the Mean Absolute Percentage Error (MAPE), with a MAPE of less than 4.56%. Xu et al. [16] introduced a feature selection method to enhance neural network training efficiency by eliminating irrelevant features during data preparation. Moreover, a skip connection is integrated into the CNN–LSTM model to mitigate neural network degradation from multi-layer LSTM. Validation using NASA and Oxford battery datasets confirms that removing less significant features improves SOH prediction accuracy while reducing neural network computational load. The CNN–LSTM-Skip model exhibits superior robustness and accuracy across varied conditions compared to other neural network models, achieving RMSE below 0.4% on both datasets. Pepe et al. [17] introduced models based on LSTM and Gated Recurrent Unit (GRU) architectures to predict both the SOH and the EOL of batteries. They trained and evaluated these models using the Massachusetts Institute of Technology (MIT) battery dataset [18], utilizing a feature representing the charge and discharge voltage summary. This feature was processed using a segmentation method based on identifying the maxima and minima points in the voltage curve, along with information about the charge policy. The models take a 68-dimensional feature as input, which is then fed into two layers of either LSTM or GRU followed by a linear layer for prediction. The LSTM model outperformed the GRU model, achieving impressive results in estimating SOH (5.49%) and EOL (1.27%). Audin et al. [19] introduced a method that utilizes auto-encoders and LSTM neural networks to forecast SOH by analyzing usage data. The efficacy

of this model was evaluated across various battery aging datasets, with the overarching goal of enhancing energy management strategies and enabling predictive maintenance for Lithium-Ion Batteries (LIBs). Similarly, Jorge et al. [20] proposed a methodology to forecast SOH evolution, leveraging features extracted from current, voltage, and temperature profiles through LSTM networks. Their investigation was conducted using datasets sourced from MIT and NASA. Liu et al. [21] introduced a method for estimating the SOH of rechargeable LIBs by harnessing the charging procedure alongside LSTM Recurrent Neural Network (LSTM-RNN) techniques. Their approach facilitated real-time SOH estimation through the utilization of health indicators derived from the charging process. Ardeshiri et al. [22] introduced an innovative deep learning strategy aimed at precisely forecasting Remaining Useful Life (RUL) through a Stacked Bidirectional LSTM (SBLSTM) model coupled with extreme gradient boosting (XGBoost). On the other hand, Toughzaou et al. [23] combined CNN with LSTM networks to estimate SOH and predict RUL for LIBs. Their model incorporated features extracted from voltage and temperature profiles (including max-value, min-value, and mean-value) using the K-means algorithm, alongside current values. This hybrid CNN–LSTM approach outperformed standalone LSTM models, providing an accurate and efficient solution for battery assessment and prognostics. From observing the latest advancements in SOH estimation (Table 1), it is evident that three primary types of models consistently emerge: CNN, LSTM networks, and Transformers. These models are often employed either independently or in combination. Typically, they aim to estimate the SOH at time $(t + w)$, where t denotes the starting time of prediction, and w represents the size of the input window. Alternatively, they may predict the SOH within the interval $[t + w, t + w + o]$, with o denoting the size of the output window. However, the iterative prediction process until reaching EOL incurs significant computational costs, as each prediction involves storing data and running models based on this data. This observation underscores the pressing need for one-shot prediction methods that can forecast the SOH until the EOL in a more resource-efficient manner. Among the existing models, only the one proposed by Li et al. [5] successfully achieves this. Therefore, we have adopted their model as a baseline to compare with our models, which provide an explanation of the decision-making process, a feature not present in the baseline model.

3. Background

This section is intended to describe the two different types of neural networks used in the present work.

3.1. LSTM model

LSTM, is a type of RNN architecture designed to effectively handle sequential data, such as time series, text, or speech. Unlike standard RNNs, which can struggle with long-term dependencies due to issues like vanishing gradients, LSTM networks are specifically structured to maintain information over longer periods. The LSTM unit contains a set of gates—input, forget, and output gates (Fig. 1), that control the flow of information in and out of the cell state, allowing the network to decide which information to keep, forget, or output at each time step. This gating mechanism helps LSTMs capture long-range dependencies in data, making them powerful for tasks like language modeling, machine translation, and time series forecasting.

3.2. Transformers

A type of neural network architecture (Fig. 2), has gained significant attention and popularity in recent years for its remarkable performance in various Natural Language Processing (NLP) tasks. Initially developed by Vaswani et al. [24], transformers have revolutionized the field of machine learning with their ability to capture long-range dependencies in sequential data.

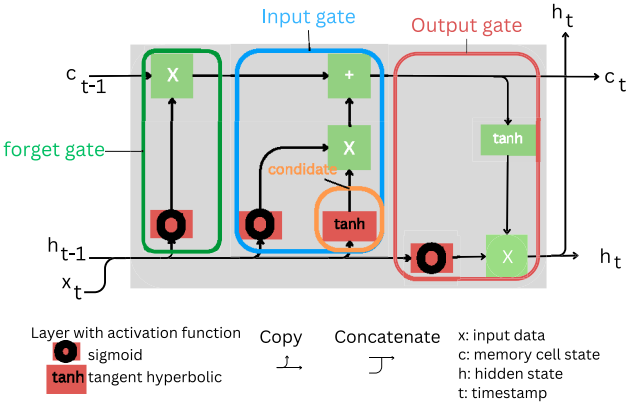

Fig. 1. LSTM cell.

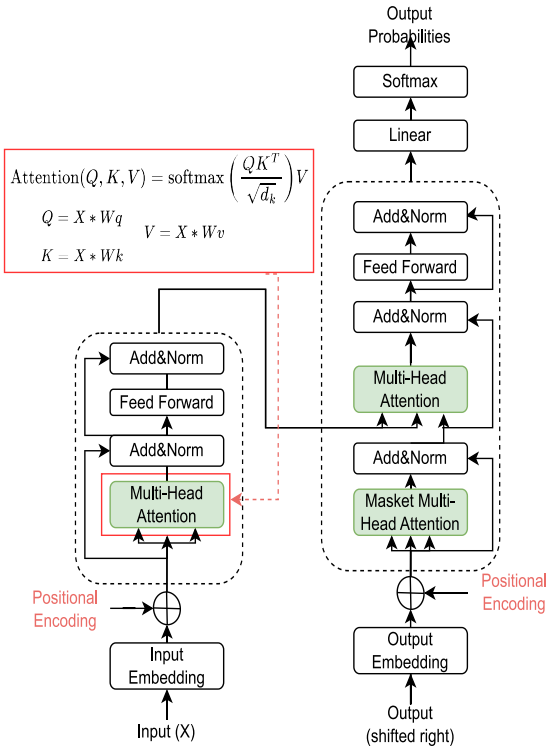

Fig. 2. The architecture of a transformer model.

As described in Fig. 2, the key components of a transformer architecture include:

- *Positional Encoding*: Since transformers do not inherently encode the order of tokens in the input sequence, positional encodings are added to the input embeddings to provide information about token positions. These can take the form of learned parameters or fixed sinusoidal functions with varying frequencies, allowing the model to distinguish between tokens based on their positions in the sequence.
- *Self-Attention Mechanism*: This mechanism allows the model to weigh the importance of each input token relative to others. For each token in the input sequence, self-attention computes three vectors sequence: query (q), key (k), and value (v). These vectors are calculated by multiplying the input sequence with three Learned Weight Matrices (LWMs), which are randomly initialized.

Table 1
Comparative analysis of SOH prediction methods.

Approach	Dataset	Model	Features/Inputs	Accuracy	One-shot SOH estimation until EOL
Xinyu et al. [11]	NASA dataset	CNN and transformer architectures	Temperature, voltage and current	RMSE less than 0.55%.	✘
Fan et al. [12]	Dataset originates from a battery performance research project at the University of Cambridge combined with another one developed by authors	T-LSTM model combining Transformer and LSTM layers	Historical voltage data	Notable reduction in error compared to simpler LSTM models with a MAE of less than 1.19%	✘
Zhang et al. [13]	Varying capacities battery datasets	LSTM neural networks	Health features from charging curves and ICA	MAE of 4.6%	✘
Wen et al. [15]	Experiments with laboratory data using LiFePO4 batteries	RBF neural network	Terminal voltage, discharge current, internal impedance combined with the state of charge	MAPE of less than 4.56% under various discharging conditions.	✘
Huanwei et al. [16]	NASA and Oxford battery datasets	CNN–LSTM-Skip	Temperature, voltage and current	RMSE below 0.4%	✘
Simona et al. [17]	MIT battery dataset	LSTM and GRU architectures	Features from charge–discharge voltage profiles	RMSE of 5.49% for SOH estimation and 1.27% for EOL estimation	✘
Audin et al. [19]	NASA and MIT	Auto-encoders and LSTM networks	Temperature, voltage and current	MAE of 1.30%	✘
Jorge et al. [20]	NASA and MIT	LSTM networks	Temperature, voltage and current	MAE of 0.77%	✘
Liu et al. [21]	NASA and Center for Advanced Life Cycle Engineering (CALCE) dataset	LSTM-RNN	Health indicators from the partial charging voltage segment and the charging current	RMSE less than 1.12%	✘
Weihan et al. [5]	Dataset generated by authors using 48 Panasonic/Sanyo UR18650E cells	Bidirectional LSTM model	Capacity time series until the present time point	MAPE of 1.1% with normal data and 1.3% with noisy data	✓
Ardeshiri et al. [22]	NASA and MIT	SBLSTM model coupled with XGBoost	Temperature, voltage and current	RMSPE of 1.94%	✘
Toughzaou et al. [23]	NASA	CNN–LSTM approach	Current associated with two values calculated using the K means algorithm from the mean, min and max values of the temperature and voltage	MAE of 0.76%	✘

The dimensions of these matrices depend on the input sequence length and the desired dimensionality of the q, k, and v vectors. During training, the model learns the optimal values for these weight matrices through backpropagation of gradients. Attention scores are calculated by taking the dot product between the q which represents a specific token, and the k vectors representing other tokens. The results are scaled down, and a softmax function is applied to emphasize higher scores. The output is then computed by multiplying the results with the v vectors.

- *Multi-Head Attention*: To enhance the model's representational capacity and capture diverse relationships between tokens, transformers employ a multi-head attention mechanism. Instead of computing attention once, multi-head attention computes attention scores multiple times in parallel, each with different learned linear projections of the input in which different LWMs are generated. The outputs from different attention heads are concatenated and linearly transformed to produce the final output.
- *Feed-Forward Neural Networks*: Following the self-attention and multi-head attention layers, transformers typically include feed-forward neural network layers to further process the token representations. These layers consist of fully connected networks with activation functions such as ReLU (Rectified Linear Unit) to introduce non-linearity and enabling the learning of complex mappings between input and output tokens.

4. Methodology for TOPS-SoHand LSTM-OSoHmodels

The proposed approach, as depicted in Fig. 3, emerged from our observation of a gap in the literature concerning the estimation of the entire SOH curve. This entails projecting the SOH of a given cell until it reaches its EOL. Two models were developed to perform this task, each serving a specific purpose.

4.1. Data preprocessing

This step is crucial for preparing the data for our models. We selected the MIT battery dataset [18] due to its comprehensive collection of over 100 cycled cells, each with varying charge policies. The charge policies are represented in the format $C1(Q1) - C2$, where $C1$ and $C2$ denote the charge capacities applied during specific charging slots, and $Q1$ represents the state of charge (SOC) at which the charging protocol transitions from $C1$ to $C2$. Table 2 provides additional details on the dataset.

From this dataset, we selected three features: current, voltage, and temperature, which serve as inputs for the models. Since the SOH cannot be directly measured and is not explicitly provided in the dataset, we calculated it based on the cell's capacity after each cycle (charge and discharge). The formula used to calculate the SOH is defined by Eq. (1) as follows:

$$SOH(\%) = \left(\frac{\text{Current capacity}}{\text{Original capacity}}\right) \times 100 \quad (1)$$

Table 2
Summary of the MIT dataset and associated testing conditions for Li-Ion Iron Phosphate (LFP)/graphite cells, including features provided.

Aspect	Details
Dataset name	MIT dataset
Collaborators	Toyota Engineering, Department of Materials Science and Engineering, Stanford University
Year of development	2019
Dataset significance	Largest available dataset regarding Li-Ion battery ageing
Cell type	Li-Ion Iron Phosphate (LFP)/Graphite
Manufacturer	A123 Systems
Model	APR18650M1A
Cell specifications	3.3 V nominal voltage 1.1 Ah nominal capacity Discharge currents up to 30 A
Testing environment	30 °C climate chamber
Discharge current	Constant current of 4.4 A (4 C)
Charging policy	Multi-step CC-CV to reduce charging time
Battery cells	124 different battery cells
Charge protocols	72 different charge protocols Detailed in Table 3
Batches	3 batches with a total size of 8.37 GB
Temperature measurement	Thermocouple attached to the surface of battery cells with thermal paste and adhesive tape
Internal resistance measurement	Measured during charging at 80% SOC Using an average of ten 30 ms or 33 ms pulses of ± 3.6
Features	Current, voltage, temperature, capacity, internal resistance and time

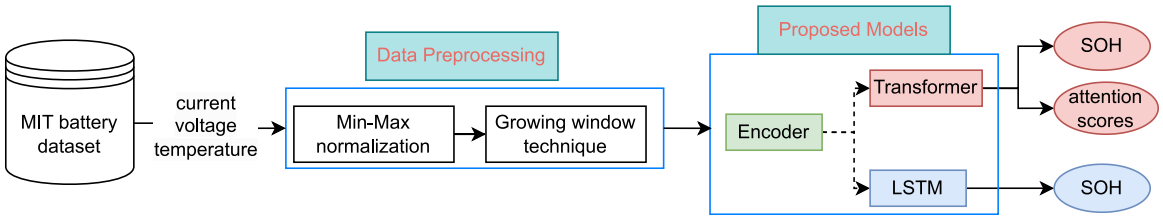

Fig. 3. Overview of the proposed approach.

To ensure consistent scaling, we applied the min–max normalization technique [25] to the inputs. Subsequently, we partitioned the data using two configurations: configuration 1, with a training-to-testing split of 60% to 40% and configuration 2 with a split of 80% to 20%. These configurations would be used to train an auto-encoder for dimensionality reduction.

Employing the growing window technique, as illustrated in Fig. 4, we fixed the input to consist of 25 cycles. This input encapsulates the values of current, voltage, and temperature across these cycles. The output comprises the SOH values of the remaining cycles until the battery reaches its EOL. With each iteration, we advanced one cycle until 10 cycles before its EOL.

This technique offers several advantages: it enables predictions using a limited number of cycles, thereby mitigating the need for storing extensive datasets. Additionally, it facilitates the estimation of SOH throughout the battery's lifespan until it reaches the EOL.

4.2. Encoder part

Recognizing the computational burden posed by the input data's dimensions, a shape of (25, 3, 1000) representing 25 cycles, each with 3 features of length 1000, we devised an auto-encoder model as an efficient solution. Employing a Dense architecture, the auto-encoder processes this voluminous data by compressing it into a lower-dimensional representation of (25, 64) before faithfully reconstructing it. The outcomes of this encoding and reconstruction procedure are succinctly outlined in Table 4 with a Mean Square Error (MSE) of less than 0.01 for both configurations. To further underscore the efficacy of our approach, we provide visual representations of the auto-encoder's ability to faithfully reconstruct the data in Fig. 5. The results demonstrate the model's capability to accurately reconstruct the original data,

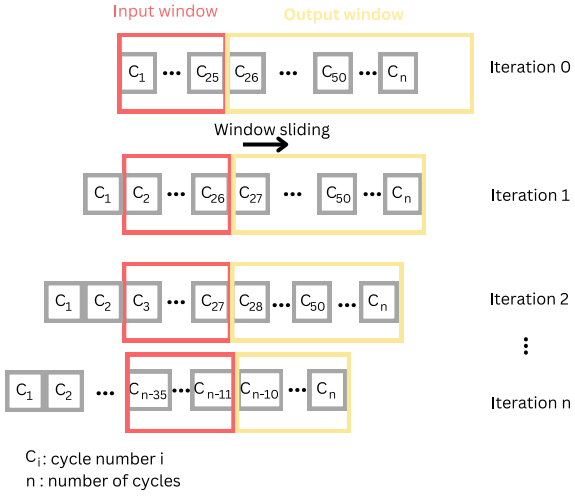

Fig. 4. The growing window technique.

particularly evident in the current and voltage variables, which exhibit similar shapes across different cells, differing primarily in width and the maximum and minimum values that reflect the respective charge and discharge policies employed. However, it is notable that the error in the model predominantly originates from the temperature feature, characterized by its complex shape.

Table 3
Charging policies and corresponding Cell IDs.

Charging policy	Cell IDs	Charging policy	Cell IDs	Charging policy	Cell IDs	Charging policy	Cell IDs
8C(35%)-3.6C	b1c44, b1c45	5.4C(60%)-3.6C	b1c16, b1c17	4.65C(44%)-5C	b2c22	4.9C(69%)-4.25C	b2c29
6C(50%)-3.6C	b1c30, b1c31	4.9C(27%)-4.75C	b2c27	8C(15%)-3.6C	b1c40, b1c41	6C(40%)-3C	b1c24, b1c25
4.4C(24%)-5C	b2c17	3.6C(30%)-6C	b2c6	4.4C(55%)-6C	b2c19	4C(40%)-6C	b2c13
6C(50%)-3C	b1c28, b1c29	5.2C(71%)-3C	b2c35	6C(52%)-3.5C	b2c46	6C(40%)-3.6C	b1c26, b1c27
5.4C(50%)-3.6C	b1c12, b1c13	4C(13%)-5C	b2c11	5.3C(54%)-4C-newstructure	b3c1, b3c9, b3c17, b3c25, b3c30, b3c34, b3c39, b3c44	2C(7%)-5.5C	b2c3
1C(4%)-6C	b2c0	5.2C(58%)-4C	b2c33	5.2C(50%)-4.25C	b2c32	2C(10%)-6C	b2c1
3.6C(2%)-4.85C	b2c5	5.6C(5%)-4.75C	b2c40	5.4C(60%)-3C	b1c14, b1c15	5.4C(40%)-3.6C	b1c8, b1c9
4.65C(19%)-4.85C	b2c21	5.6C(58%)-3.5C	b2c39	5.6C(38%)-4.25C	b2c37	5.4C(70%)-3C	b1c18, b1c19
7C(40%)-3.6C	b1c38, b1c39	5.2C(66%)-3.5C	b2c34	5.6C(36%)-4.3C-newstructure	b3c3, b3c5, b3c12, b3c14, b3c19, b3c27, b3c36, b3c41	8C(25%)-3.6C	b1c42, b1c43
3.7C(31%)-5.9C-newstructure	b3c6, b3c21, b3c28	3.6C(80%)-3.6C	b1c0, b1c1, b1c2	5.9C(60%)-3.1C-newstructure	b3c22, b3c31	4.8C(80%)-4.8C	b1c6, b1c7, b2c24, b2c25, b2c26
5.2C(37%)-4.5C	b2c31	5.6C(19%)-4.6C-newstructure	b3c2, b3c4, b3c11, b3c13, b3c18, b3c26, b3c35, b3c40	4.8C(80%)-4.8C-newstructure	b3c7, b3c10, b3c16, b3c23, b3c32, b3c37, b3c42, b3c45	5.4C(50%)-3C	b1c10, b1c11
3.6C(9%)-5C	b2c10	6C(30%)-3.6C	b1c22, b1c23	5.4C(80%)-5.4C	b1c20, b1c21	5C(67%)-4C-newstructure	b3c0, b3c8, b3c20, b3c24, b3c33, b3c38, b3c43
4.4C(8%)-4.85C	b2c20	4C(4%)-4.85C	b2c14	4.4C(47%)-5.5C	b2c18	4.4C(80%)-4.4C	b1c5
4.9C(61%)-4.5C	b2c28	5.2C(10%)-4.75C	b2c30	5.6C(25%)-4.5C	b2c36	XC(80%)-4C	b2c15, b2c16
6C(31%)-4.25C	b2c43	2C(2%)-5C	b2c2	6C(40%)-4C	b2c44	7C(30%)-3.6C	b1c34, b1c35
4.65C(69%)-6C	b2c23	XC(80%)-3.6C	b2c7, b2c8, b2c9	6C(60%)-3C	b1c32, b1c33, b2c47	4C(80%)-4C	b1c3, b1c4
7C(40%)-3C	b1c36, b1c37	5.6C(47%)-4C	b2c38	5.6C(65%)-3C	b2c41	6C(4%)-4.75C	b2c45
4C(31%)-5	b2c12	6C(20%)-4.5C	b2c42	3.6C(22%)-5.5C	b2c4	5.9C(15%)-4.6C-newstructure	b3c15, b3c29

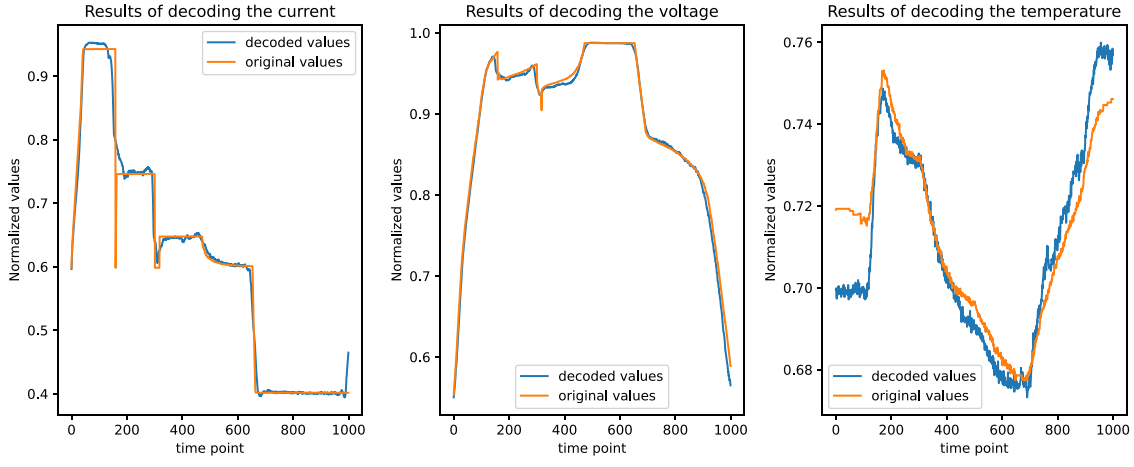

Fig. 5. Visual representation results of the data decoding for the current, voltage and temperature, respectively.

4.3. TOPS-SoHmodel

We propose two models with nearly identical architectures, as illustrated in Fig. 6. The primary distinction between them lies in how the attention scores are calculated.

The TOPS-SoH$_i$ model calculates attention scores solely between input cycles, aiming to identify potential inter-cycle relationships. Conversely, the TOPS-SoH$_{io}$ model computes attention scores between inputs and outputs, aiming to discern which cycles significantly influence predictions. This is achieved by incorporating a multi-head

Table 4
Comparison of MAE and MSE values of the two proposed configurations (error values in 10^{-2}).

Configuration	Splitting	MAE	MSE
Configuration 1	Training (60%)	0.96	0.14
	Testing (40%)	1.06	0.13
Configuration 2	Training (80%)	0.86	0.13
	Testing (20%)	1.09	0.16

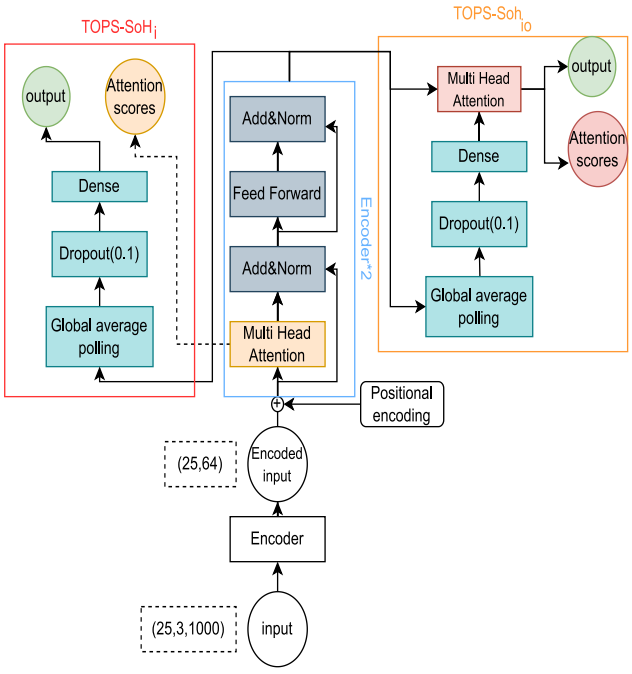

Fig. 6. The architecture of TOPS-SoH(i and io) models.

attention layer with context (the Query) derived from the inputs and the Key and Value sourced from the Dense layer representing the output.

The input to these models is derived from the encoder model output obtained from the auto-encoder step. To encode positional information, we employ position encoding, aiding in identifying the position of each cycle in the sequence, as described by Eqs. (2) and (3).

$$\mathrm{PE}(pos, 2i) = \sin\left(\frac{pos}{10000^{2i/512}}\right) \tag{2}$$

$$\mathrm{PE}(pos, 2i+1) = \cos\left(\frac{pos}{10000^{2i/512}}\right) \tag{3}$$

where:

- pos : represents the position of the cycle in the sequence,
- i : is the position of the value in the encoded vector.
- This term ($10000^{\frac{2i}{d_{\text{model}}}}$) scales the input position (pos), ensuring different dimensions use different frequencies.
- The use of sine and cosine functions introduces periodic patterns, allowing the model to learn relative positions and distances between tokens.

As illustrated in Fig. 6, we construct a stack of encoders. Each encoder comprises a multi-head attention layer to capture inter-cycle dependencies, a feed-forward layer to capture intricate dependencies, and a normalization layer to standardize the data, considering potential alterations in data range post-addition. To address variations in the number of cycles among cells, zero-padding is employed to ensure uniform data shape consistency. To prevent the model from considering padding during training, we adapt the loss function accordingly, as outlined in Algorithm 1. The algorithm begins by identifying non-zero elements in the original data using the *not_equal* function. The masked error is then computed, focusing solely on non-zero elements. Finally, the result is obtained by summing the masked errors and dividing by the sum of the mask elements using the *reduce_sum* function.

Algorithm 1 Masked Mean Absolute Error (MMAE) Loss

Require: y_{true}: Original values
Require: y_{pred}: Predicted values
1: $mask \leftarrow \text{not\_equal}(y_{\text{true}}, 0)$
2: $masked\_error \leftarrow |\, y_{\text{true}} - y_{\text{pred}} \,| \times mask$ ▷ MMSE:
$masked\_error = (y_{\text{true}} - y_{\text{pred}})^2 \times mask$
3: $masked\_mae \leftarrow \frac{\text{reduce\_sum}(masked\_error)}{\text{reduce\_sum}(mask)}$
4: **return** $masked\_mae$

The SOH estimation was done using Global average polling and Dense layers, chosen to reduce the number of trainable parameters and as consequence the size of the model. Both models underwent training and testing using the K-fold cross-validation technique [26] to mitigate overfitting and to assess performance across various training and testing sets. They were compiled using the Adam optimization method with a learning rate of 10^{-4}.

4.4. LSTM-OSoHmodel

The LSTM-OSoHmodel presents a distinctive approach compared to the previously discussed TOPS-SoHmodels, leveraging LSTM cells for sequence modeling. The LSTM-OSoHarchitecture is characterized by a single layer of LSTM cells containing 512 nodes and utilizes the hyperbolic tangent (tanh) activation function. Following the LSTM layer, there is a Dense layer employing a linear activation function. This design aims to capture temporal dependencies within the input sequence effectively.

Similar to the TOPS-SoHmodels, the LSTM-OSoHmodel undergoes training and testing using the same methodologies. It is trained on the available dataset using techniques such as K-fold cross-validation, ensuring robustness in performance evaluation and mitigating overfitting concerns, with k set to five. The Adam optimization method is employed for model compilation, with a learning rate set at 10^{-4}.

5. Results and discussion

The models were developed, trained, and tested using the TensorFlow framework on a high-performance machine equipped with dual Intel Xeon E5-2680 v4 CPUs, each with 14 cores and a total of 56 threads across the system, operating at a base frequency of 2.40 GHz and capable of turbo-boosting up to 3.3 GHz. The system featured 251 GB of RAM, providing ample memory for handling and processing large datasets, and a large shared L3 cache of 35.84 MB, which facilitated efficient data access during computations. Although the node lacked a GPU, the powerful CPU configuration, combined with a fast 512 GB SSD for quick data retrieval and storage, ensured that the models could be trained and tested efficiently. The system's Non-uniform memory access (NUMA) architecture, with two nodes, further optimized memory access, contributing to the overall performance. In summary, this machine's robust CPU, substantial memory, and high-speed storage, along with the TensorFlow framework, enabled us to effectively develop, train, and test complex models, even in the absence of dedicated GPU acceleration.

To evaluate the performance of the proposed models, we examine their estimations across three distinct time slots:

- *Earliest*: when the prediction starts at the beginning of life.
- *Intermediate*: when the prediction starts at the middle of life.

Table 5
Comparison of MMAE, MMSE (error values in 10^{-2}), and MAPE values for the proposed and state-of-the-art models (associated with their storage size, number of parameters, and training time) across the Earliest, Intermediate, and Latest time slots.

Model	Params	Training time (h)	Storage (MB)	Metric	Earliest		Intermediate		Latest	
					Mean	STD	Mean	STD	Mean	STD
E-Transformer$_i$	**669k**	**0.033**	**8.17**	MMAE	1.62	1.18	1.21	0.77	1.89	1.90
				MMSE	0.06	0.13	0.04	0.05	0.09	0.22
				MAPE	2.35	1.83	2.16	1.50	2.73	2.44
E-Transformer$_{io}$	805k	0.676	9.6	MMAE	2.43	1.54	2.33	1.21	2.46	1.74
				MMSE	0.12	0.16	0.11	0.11	0.12	0.20
				MAPE	2.58	1.72	2.38	1.31	2.65	2.06
E-LSTM	2.20M	0.037	26.5	MMAE	**0.77**	**0.59**	**0.62**	**0.46**	**0.71**	**0.66**
				MMSE	**0.02**	**0.04**	**0.01**	**0.02**	**0.13**	**0.04**
				MAPE	**1.70**	**1.52**	**1.81**	**1.58**	**1.93**	**1.61**
BiLSTM [5]	2.28M	3.122	26.9	MMAE	3.35	0.62	3.23	1.55	6.73	2.69
				MMSE	0.18	0.11	0.24	0.21	0.68	0.45
				MAPE	3.62	0.78	3.66	1.86	7.76	3.30

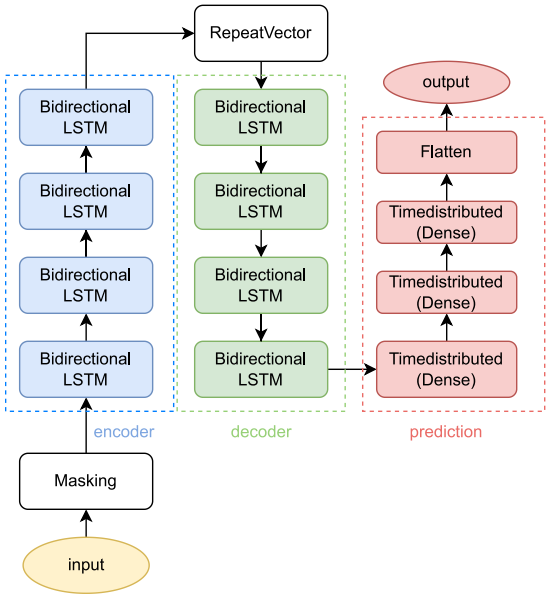

Fig. 7. The bidirectional LSTM model architecture [5].

- *Latest*: when the prediction starts at the end of life.

Table 5 presents the MMAE, Masked Mean Square Error (MMSE) and MAPE for the proposed models across the three-time slots. The table showcases both the Mean and Standard Deviation (STD) of these metrics for each model at each time slot, providing insights into their performance consistency. Additionally, it includes details on the number of trainable parameters, training time and the storage size for each model, highlighting the computational complexity and memory requirements associated with their implementation. The models were compared with the model proposed by Li et al. [5], which uses a stack of Bidirectional LSTM (BiLSTM) layers to encode and then decode the data. The architecture of this model is illustrated in Fig. 7.

Figs. 10 to 12 depict the performance of the models using data from three randomly selected cells, each from a different batch. These figures illustrate the models' ability to estimate the SOH until the EOL of the cells, irrespective of the number of cycles each cell has undergone. Notably, the performance is observed across cells with varying numbers of cycles, ranging from fewer to greater cycles. Regarding the attention scores, as discussed earlier and taking the *Earliest* estimation for cell b3c7 as an example, two attentions were calculated. One is for the *TOPS-SoH*$_i$ model, which represents the attention between input cycles as illustrated in Fig. 8, resulting in a 25 by 25 matrix indicating

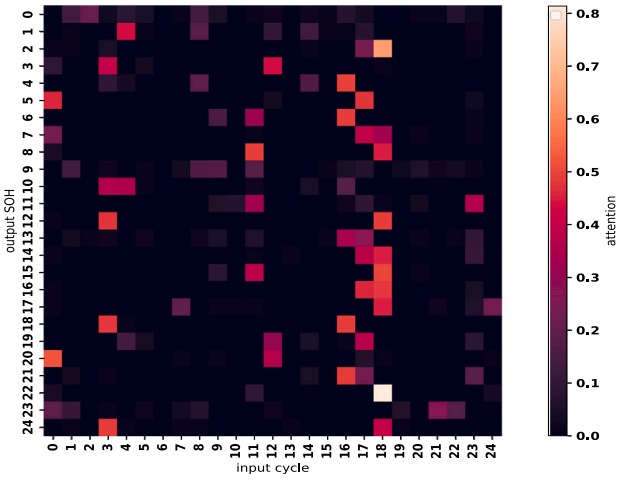

Fig. 8. Attention scores of *TOPS-SoH*$_i$ model.

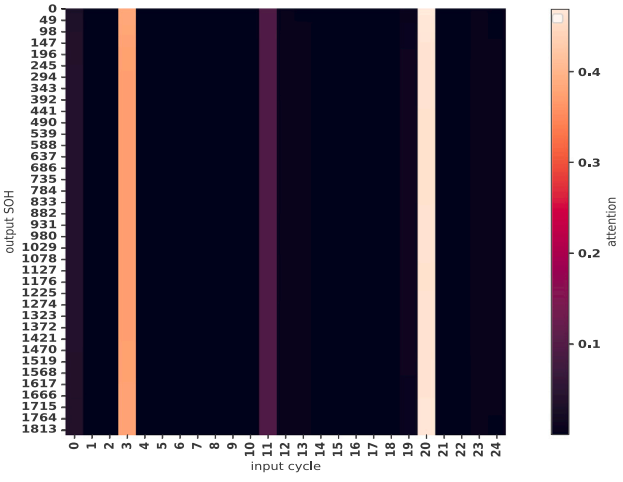

Fig. 9. Attention scores of *TOPS-SoH*$_{io}$ model.

the relationships between them. For instance, we can observe strong attention between cycle number 18 and the other cycles, especially with cycle number 22. On the other hand, the attention scores of *TOPS-SoH*$_{io}$ model reflect the relationship between the inputs and the outputs, as shown in Fig. 9, indicating the cycles that affect the prediction. This is important for understanding how the model makes predictions. In this example, it showcases that the model uses cycles 3, 11, and 20 to make

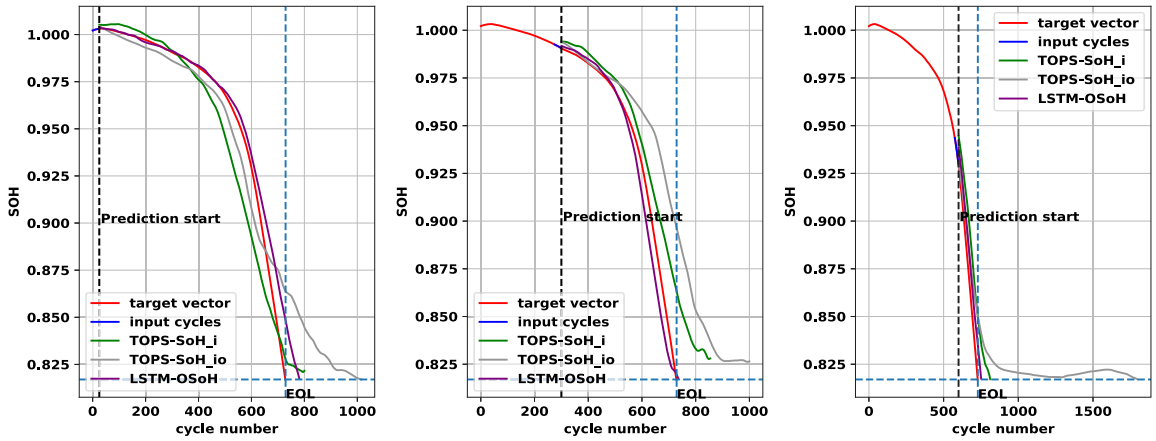

Fig. 10. Comparison of proposed models prediction for cell b1c32 across the three distinct time slots.

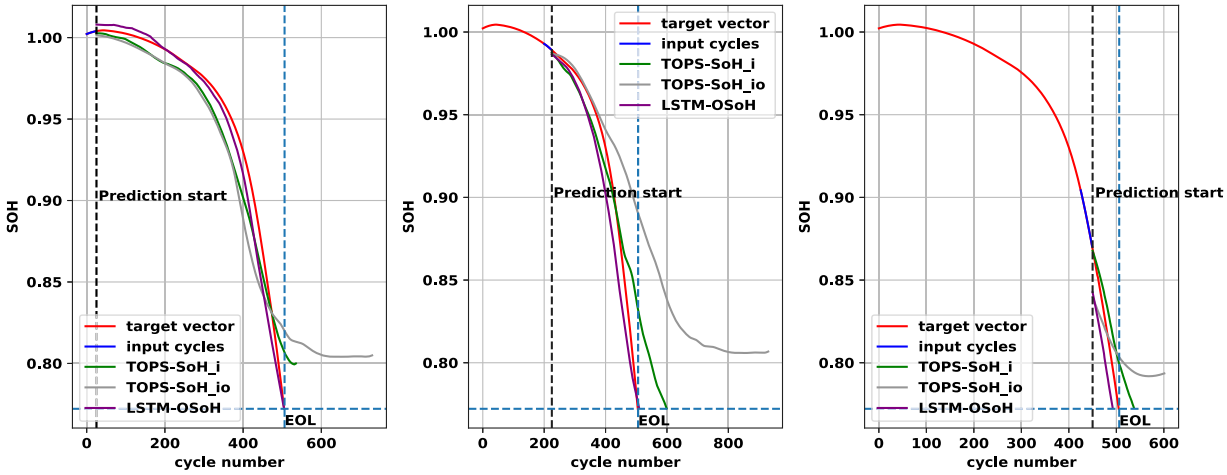

Fig. 11. Comparison of proposed models prediction for cell b2c5 across the three distinct time slots.

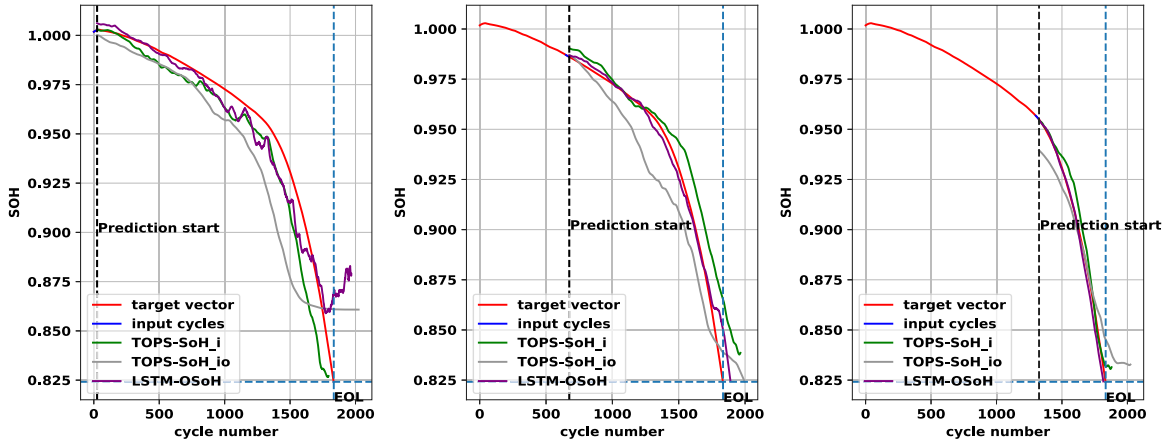

Fig. 12. Comparison of proposed models prediction for cell b3c7 across the three distinct time slots.

the prediction. After analyzing the data contained within these cycles, we found that they are close to the average across the three features, reflecting the fact that we are using global average pooling to make the estimation. Based on the results obtained, we observed that the *LSTM-OSoH* model outperformed the other models in terms of accuracy, with an MMAE of less than 0.01 across all time slots. This makes it an ideal choice for accurate SOH estimation. However, its limitations include higher parameter count and larger storage requirements, making it less suitable for resource-constrained environments like embedded BMS. On the other hand, the *TOPS-SoH* models, especially the *io* variant, demonstrate acceptable accuracy compared to the *BiLSTM* model [5]. These models not only match up in terms of accuracy but also excel in their ability to elucidate the cycles used for estimation and provide detailed attention to each SOH estimation value. Moreover, they excel in terms of the number of trainable parameters and storage size, making them highly suitable for integration into the BMS. A notable feature of

the proposed models is their ability to estimate the RUL of the batteries. By simply analyzing the length of the output of the models, we can predict the number of remaining cycles until the battery reaches its EOL.

6. Conclusion

In this study, we compared the performance of various models in estimating the SOH until reaching the EOL. Our findings indicate that the *LSTM-OSoH* model consistently outperformed other models in terms of accuracy, boasting an MMAE of less than 0.01 across all time slots (earliest, intermediate, and latest). Specifically, the *LSTM-OSoH* achieved MMAE values of 0.77% (earliest), 0.62% (intermediate), and 0.71% (latest), along with MMSE values as low as 0.01%. This high level of precision, combined with a low MAPE of around 1.70%, makes it a robust choice for applications that prioritize accuracy, without constraints on storage or the necessity for an in-depth understanding of the model's internal mechanics.

However, the *TOPS-SoH* models, particularly the *io* variant, offer a compelling alternative. While they may not achieve the same level of accuracy as the *LSTM-OSoH* model, they provide acceptable performance compared to baseline models like the *BiLSTM*. For instance, the $TOPS\text{-}SoH_{io}$ model achieved an MMAE of 2.43% (earliest), 2.33% (intermediate), and 2.46% (latest), with corresponding MMSE values of approximately 0.11%. Moreover, the MAPE values for this model were within the range of 2.58% to 2.65%. Although these figures are slightly higher than those of the *LSTM-OSoH*, they still represent a significant improvement over the *BiLSTM*, which recorded MMAE values as high as 6.73% in the latest time slot and MAPE values reaching 7.76%.

Additionally, the *TOPS-SoH* models excel in highlighting the cycles used for estimation and providing detailed attention to each SOH estimation value. The efficiency in terms of trainable parameters and storage size makes them particularly attractive for integration into BMS systems where resource constraints are a concern.

In conclusion, the choice of SOH estimation model depends on the specific requirements of the application. For scenarios where accuracy is paramount and resource constraints are not an issue, the *LSTM-OSoH* model emerges as the preferred choice, given its superior performance across all metrics. Conversely, for applications valuing efficiency and interpretability, the *TOPS-SoH* models, with their balance of performance and efficiency, offer a promising solution for integration into BMS systems. Future studies could concentrate on refining these models specifically for different battery types, aiming to improve their accuracy and usefulness in practical situations.

CRediT authorship contribution statement

Slimane Arbaoui: Writing – original draft, Visualization, Methodology, Formal analysis, Conceptualization. **Ahmed Samet:** Writing – review & editing, Supervision. **Ali Ayadi:** Writing – review & editing, Supervision. **Tedjani Mesbahi:** Writing – review & editing, Supervision. **Romuald Boné:** Writing – review & editing, Supervision.

Declaration of competing interest

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

Data availability

Data will be made available on request.

Acknowledgments

This research received partial funding from the French National Research Agency (ANR) under the project "ANR-22-CE92-0007-02". Additionally, support was provided by the European Union through the Horizon Europe program and the innovation program under "GAP-101103667".

Code availability

The code for implementing the approach described in this paper is accessible at https://github.com/arslimane/Energetic-project-2.git.

References

[1] Chuang Yu-Sen, Cheng Hong-Ping, Cheng Chin-Chi. Reuse of retired lithium-ion batteries (LIBs) for electric vehicles (EVs) from the perspective of extended producer responsibility (EPR) in Taiwan. World Electr Veh J 2024;15.

[2] Zhao Xu, Hu Jianyao, Hu Guangdi, Qiu Huimin. A state of health estimation framework based on real-world electric vehicles operating data. J Energy Storage 2023;63:107031.

[3] Vertiv. Implementing proactive battery management strategies. 2023, https://www.vertiv.com/en-us/about/news-and-insights/articles/educational-articles/implementing-proactive-battery-management-strategies/. [Accessed 07 June 2024].

[4] Saqli Khadija, Bouchareb Houda, Oudghiri Mohamed, M'Sirdi Nacer K. An overview of state of charge(soc) and state of health(soh) estimation methods of li-ion batteries. 2019, URL https://api.semanticscholar.org/CorpusID:212631016.

[5] Li Weihan, Sengupta Neil, Dechent Philipp, Howey David, Annaswamy Anuradha, Sauer Dirk Uwe. One-shot battery degradation trajectory prediction with deep learning. J Power Sources 2021;506:230024.

[6] Xu Cai, Zhao Wei, Zhao Jinglong, Guan Ziyu, Song Xiangyu, Li Jianxin. Uncertainty-aware multiview deep learning for internet of things applications. IEEE Trans Ind Inf 2023;19(2):1456–66.

[7] Darius Roman, Saurabh Saxena, Valentin Robu, Michael Pecht, David Flynn. Machine learning pipeline for battery state-of-health estimation. Nat Mach Intell 2021;3:447–56.

[8] Neupert Steven, Kowal Julia. Model-based state-of-charge and state-of-health estimation algorithms utilizing a new free lithium-ion battery cell dataset for benchmarking purposes. Batteries 2023;9(7).

[9] He Hongwen, Xiong Rui, Fan Jinxin. Evaluation of lithium-ion battery equivalent circuit models for state of charge estimation by an experimental approach. Energies 2011;4(4):582–98.

[10] Song Ke, Huang Xing, Huang Pengyu, Sun Hui, Chen Yuhui, Huang Dongya. Data-driven health state estimation and remaining useful life prediction of fuel cells. Renew Energy 2024;120491.

[11] Gu Xinyu, See KW, Li Penghua, Shan Kangheng, Wang Yunpeng, Zhao Liang, Lim Kai Chin, Zhang Neng. A novel state-of-health estimation for the lithium-ion battery using a convolutional neural network and transformer model. Energy 2023;262:125501.

[12] Fan Yuqian, Li Yi, Zhao Jifei, Wang Linbing, Yan Chong, Wu Xiaoying, Zhang Pingchuan, Wang Jianping, Gao Guohong, Wei Liangliang. Online state-of-health estimation for fast-charging lithium-ion batteries based on a transformer–long short-term memory neural network. Batteries 2023;9(11).

[13] Zhang Hao, Sun Hanlei, Kang Le, Zhang Yi, Wang Licheng, Wang Kai. Prediction of health level of multiform lithium sulfur batteries based on incremental capacity analysis and an improved LSTM. Protect Control Modern Power Syst 2024;9(2):21–31.

[14] Kingma Diederik P, Ba Jimmy. Adam: A method for stochastic optimization. 2014, arXiv preprint arXiv:1412.6980.

[15] Chang Wen-Yeau, Chang Po-Chuan. Application of radial basis function neural network, to estimate the state of health for LFP battery. Int J Electr Electron Eng (IJEEE) 2017;7:1–6.

[16] Xu Huanwei, Wu Lingfeng, Xiong Shizhe, Li Wei, Garg Akhil, Gao Liang. An improved CNN-LSTM model-based state-of-health estimation approach for lithium-ion batteries. Energy 2023;276:127585.

[17] Pepe Simona, Ciucci Francesco. Long-range battery state-of-health and end-of-life prediction with neural networks and feature engineering. Appl Energy 2023;350:121761.

[18] Severson Kristen A, Attia Peter M, Jin Norman, Perkins Nicholas, Jiang Benben, Yang Zi, Chen Michael H, Aykol Muratahan, Herring Patrick K, Fraggedakis Dimitrios, Bazant Martin Z, Harris Stephen J, Chueh William C, Braatz Richard D. Data-driven prediction of battery cycle life before capacity degradation. Nature Energy 2019;383–91.

[19] Audin Paul, Jorge Inès, Mesbahi Tedjani, Samet Ahmed, Bertr François De, Boné De Beuvronand Romuald. Auto-encoder LSTM for li-ion SOH prediction : a comparative study on various benchmark datasets. In: 2021 20th IEEE international conference on machine learning and applications. ICMLA, Vol. 8, 2021, p. 1526–36.

[20] Jorge Inès, Mesbahi Tedjani, Samet Ahmed, Boné Romuald. Time series feature extraction for lithium-ion batteries state-of-health prediction. J Energy Storage 2023;59.

[21] Liu Kang, Kang Longyun, Xie Di. Online state of health estimation of lithium-ion batteries based on charging process and long short-term memory recurrent neural network. In: Batteries 2023 MDPI. Vol. 16, 2023, p. 1–16.

[22] Ardeshiri Reza Rouhi, Liu Ming, Ma Chengbin. Multivariate stacked bidirectional long short term memory for lithium-ion battery health management. Reliab Eng Syst Saf 2022;224:108481.

[23] Toughzaoui Yassine, Toosi Safieh Bamati, Chaoui Hicham, Louahlia Hasna, Petrone Raffaele, Masson Stéphane Le, Gualous Hamid. State of health estimation and remaining useful life assessment of lithium-ion batteries: A comparative study. J Energy Storage 2022;51.

[24] Vaswani Ashish, Shazeer Noam, Parmar Niki, Uszkoreit Jakob, Jones Llion, Gomez Aidan N, Kaiser Łukasz, Polosukhin Illia. Attention is all you need. Adv Neural Inf Process Syst 2017;30.

[25] Patro SGOPAL, Sahu Kishore Kumar. Normalization: A preprocessing stage. 2015, arXiv preprint arXiv:1503.06462.

[26] Anguita Davide, Ghelardoni Luca, Ghio Alessandro, Oneto Luca, Ridella Sandro, et al. The'k'in K-fold cross validation. In: ESANN. Vol. 102, 2012, p. 441–6.