Università degli Studi di Genova
Scuola Politecnica & Facoltà di Ingegneria

Laurea Magistrale in Ingegneria Strategica

International MSc in Engineering Technology for Strategy and Security of Genoa University

---

Deep Learning-Based State of Charge and State of Health Estimation for Lithium-Ion Batteries

---

Advisor: **Prof. Francesco Bellotti**

Co-Advisor: **Dr. Matteo Fresta**

Candidate: **Hamed Zanganeh**

Academic Year 2025–2026

STRATEGOS 1st Edition

Abstract

This thesis investigates data-driven estimation of two critical battery health indicators—State of Charge (SoC) and State of Health (SoH)—for lithium-ion batteries using deep learning. Both studies utilise the comprehensive aging dataset published by Luh and Blank (Scientific Data, 2024), which contains measurements from 228 commercial LG INR18650HG2 NMC/C+SiO cells (3.0 Ah nominal capacity) aged under 76 different operating conditions encompassing calendar aging, cyclic aging, and various driving profiles, with temperatures ranging from 0 °C to 40 °C.

For SoC estimation, four neural network architectures—Multi-Layer Perceptron (MLP), Convolutional Neural Network (CNN), Long Short-Term Memory (LSTM), and Gated Recurrent Unit (GRU)—are systematically compared. The models receive only voltage, current, and temperature as inputs within a fixed-length window of 64 time steps (approximately 21.3 minutes at 20-second resolution) and predict the SoC at a forecast horizon of 8 steps ahead. Data from cells with SoH ≥ 95% is used. The GRU achieves the best test-set performance with an RMSE of 0.9513 percentage points and an MAE of 0.1614 percentage points, followed closely by the LSTM (RMSE = 1.0040), CNN (RMSE = 1.0343), and MLP (RMSE = 1.2720). All models exceed $R^2 = 0.998$.

For SoH estimation, three deep sequence learning architectures—LSTM, GRU, and a 1D Convolutional Neural Network (CNN1D)—are evaluated. The models process 256-timestep windows (approximately 8.5 hours) of six physical features: voltage, current, temperature, open-circuit voltage, dV/dt, and dI/dt. A strict profile-level data split ensures that entire aging conditions are held out as unseen test data. The CNN1D emerges as the best-performing model with an MAE of 3.31%, an RMSE of 5.65%, and an R^2 of 0.812 on completely unseen test profiles. The GRU achieves an MAE of 3.85% ($R^2 = 0.770$) and the LSTM an MAE of 4.31% ($R^2 = 0.728$).

Together, the SoC and SoH studies demonstrate that gated recurrent architectures excel at short-horizon, high-precision SoC tracking, while hierarchical convolutional architectures prove more robust for the longer-horizon, cross-condition SoH prediction task. The findings provide practical

guidance for deploying deep learning models in battery management systems.

Keywords

Lithium-ion battery; State of Charge; State of Health; deep learning; LSTM; GRU; CNN; battery aging; Battery Management System; LG INR18650HG2

Acknowledgements

I would like to express my sincere gratitude to my supervisor for their guidance, support, and valuable feedback throughout the development of this thesis. Their comments and encouragement helped shape both the technical direction and the final presentation of this work.

I would also like to thank the authors and maintainers of the publicly available battery aging dataset used in this thesis. Their effort in collecting, documenting, and publishing a large-scale and carefully structured experimental dataset made this research possible.

Finally, I would like to thank my family and friends for their patience, encouragement, and support during the completion of this work.

Abbreviations

Ah — Ampere-hour

BMS — Battery Management System

CC — Constant Current

CC-CV — Constant Current–Constant Voltage

CNN — Convolutional Neural Network

CNN1D — One-Dimensional Convolutional Neural Network

CV — Constant Voltage

DoD — Depth of Discharge

ECM — Equivalent Circuit Model

EFC — Equivalent Full Cycles

EIS — Electrochemical Impedance Spectroscopy

EKF — Extended Kalman Filter

EoL — End of Life

GRU — Gated Recurrent Unit

ICA — Incremental Capacity Analysis

LIB — Lithium-Ion Battery

LSTM — Long Short-Term Memory

MAE — Mean Absolute Error

MAPE — Mean Absolute Percentage Error

ML — Machine Learning

MLP — Multi-Layer Perceptron

NCA — Nickel Cobalt Aluminum

NMC — Nickel Manganese Cobalt

OCV — Open Circuit Voltage

RMSE — Root Mean Square Error

SEI — Solid Electrolyte Interphase

SoC — State of Charge

SoH — State of Health

SVR — Support Vector Regression

Nomenclature

V(t) — terminal voltage as a function of time

I(t) — current as a function of time

T(t) — temperature as a function of time

OCV(t) — estimated open-circuit voltage

$Q_{nominal}$ — nominal battery capacity

$Q_{current}$ — current usable capacity

$Q_{available}$ — currently available usable capacity

SoC — State of Charge

SoH — State of Health

ŷ — predicted target value

y — true target value

T — sequence length

F — number of features

μ — mean used in normalization

σ — standard deviation used in normalization

R^2 — coefficient of determination

L_{MSE}— mean squared error loss

Table of Contents

List of Figures

List of Tables

Chapter 1 – Introduction

1.1 Motivation and Context

Lithium-ion batteries (LIBs) have become the dominant energy storage technology for electric vehicles (EVs), stationary grid storage, and consumer electronics. The global Li-ion battery market is projected to exceed 4,700 GWh of annual capacity by 2030, driven by the worldwide push toward electrification and decarbonisation. At the core of every Battery Management System (BMS) lies the accurate estimation of the battery's internal states, most critically the State of Charge (SoC) and the State of Health (SoH).

SoC represents the remaining usable energy in a battery relative to its full capacity and directly influences decisions related to energy management, charge control, and operational safety. Conventional model-based approaches, such as Coulomb counting and equivalent circuit models, are sensitive to measurement noise, cumulative integration errors, and require precise parameterisation that may drift with cell aging.

SoH, defined as the ratio of the current maximum usable capacity to the original rated capacity, enables critical decisions such as range prediction in EVs, warranty assessment, second-life repurposing, and predictive maintenance scheduling. However, SoH cannot be measured directly from any single sensor; it must be inferred from indirect measurable quantities such as terminal voltage, current flow, cell temperature, and impedance characteristics. This inference challenge makes SoH estimation one of the most actively researched problems in battery science.

Data-driven methods based on deep learning have emerged as promising alternatives for both SoC and SoH estimation, capable of learning complex nonlinear mappings between measurable electrical quantities and the internal state of the cell without explicit electrochemical modelling. This thesis presents a systematic investigation of deep learning architectures for both estimation tasks, using a single comprehensive dataset.

1.2 Research Objectives

The specific objectives of this thesis are:

- SoC estimation: To compare four neural network architectures—MLP, CNN, LSTM, and GRU—for SoC estimation using only voltage, current, and temperature as inputs, without any prior SoC information.
- SoH estimation: To evaluate three deep sequence learning architectures—LSTM, GRU, and CNN1D—for SoH prediction from 256-timestep windows of six physical features, under a strict profile-level data split.
- Comparative analysis: To draw cross-cutting insights about which architectural families are best suited for short-horizon SoC tracking versus longer-horizon SoH prediction.
- Practical guidance: To provide actionable recommendations for deploying deep learning models in battery management systems.

1.3 Thesis Structure

This thesis is organised as follows. Chapter 2 provides the necessary background on battery fundamentals and degradation physics. Chapter 3 describes the dataset and experimental context. Chapter 4 presents the SoC estimation methodology. Chapter 5 reports the SoC results and discussion. Chapter 6 details the SoH estimation methodology. Chapter 7 presents the SoH results and discussion. Chapter 8 provides a comparative discussion integrating both studies. Chapter 9 concludes the thesis with a summary and directions for future work.

Chapter 2 – Battery Fundamentals and Degradation Physics

2.1 Lithium-Ion Battery Operating Principles

Lithium-ion batteries store and release electrical energy through the reversible intercalation of lithium ions between two host electrodes separated by an electrolyte. During discharge, lithium ions migrate from the anode (negative electrode) through the electrolyte to the cathode (positive electrode), while electrons flow through the external circuit to provide electrical power. During charge, an external potential reverses this process, driving lithium ions back into the anode.

The cell voltage is determined by the difference in electrochemical potential between the cathode and anode materials and varies with the degree of lithiation (state of charge), temperature, and current magnitude. Key measurable quantities include the terminal voltage, the current flowing through the cell, the cell surface temperature, and the open-circuit voltage (OCV), which is the equilibrium voltage at zero current after sufficient relaxation.

2.2 State of Charge (SoC)

The State of Charge quantifies the remaining usable energy in the battery as a fraction or percentage of its full capacity. It is analogous to a fuel gauge for conventional vehicles. SoC is typically defined on a scale from 0% (fully discharged) to 100% (fully charged). Accurate SoC estimation is essential for range estimation, energy management, and protection against over-charge or deep discharge conditions that can damage the cell.

Traditional SoC estimation methods include Coulomb counting (integration of current over time), which suffers from cumulative drift errors; OCV-based methods, which require long rest periods for voltage relaxation; and model-based approaches using Kalman filters applied to equivalent circuit models, which require accurate parameterisation. Data-driven approaches aim to learn the SoC directly from measurable inputs without these limitations.

2.3 State of Health (SoH)

The State of Health reflects the degree of degradation of the battery relative to its original condition. SoH is commonly defined in terms of capacity fade:

$$SoH = \frac{Q_{aged}}{Q_{nominal}} \times 100\%$$

where Q_aged is the measured discharge capacity under standardised conditions and Q_nominal is the nominal capacity at the beginning of life. A battery is typically considered to have reached its end of life when the SoH drops below 80% in automotive applications or 70% in less critical applications.

Traditional SoH estimation methods include capacity-based methods (requiring full charge-discharge cycles), incremental capacity analysis (ICA/DVA) which tracks phase transitions through dQ/dV curves but requires slow-rate cycling, and electrochemical impedance spectroscopy (EIS) which provides frequency-domain impedance information but requires specialised instrumentation.

2.4 The LG INR18650HG2 Cell

The cells studied in this work are LG INR18650HG2, a commercial 18650-format cylindrical cell with the following properties (based on the manufacturer's technical specification and the Luh and Blank 2024 dataset):

Table 1: Properties of the LG INR18650HG2 cell.

Parameter	Value
Form factor	18650 (cylindrical)
Nominal capacity	3000 mAh (3.0 Ah)
Nominal voltage	3.60 V
Usable energy	11.0 Wh
Allowed voltage range	2.0–4.2 V (used: 2.5–4.2 V)
Cathode material	$LiNi_xMn_yCo_{1-x-y}O_2$ (NMC)
Anode material	Graphite with SiO (C+SiO)
Maximum current	6.0 A (charge) / 20.0 A (discharge)
Energy density	240 Wh/kg, 640 Wh/L
Temperature range	−20 to +75 °C (operation), −5 to +50 °C (charging)

2.5 Degradation Mechanisms in NMC/C+SiO Cells

The NMC cathode paired with a Carbon-Silicon oxide (C-SiO) blended anode achieves among the highest volumetric energy densities available in the 18650 cylindrical format. However, the incorporation of silicon in the anode introduces complex degradation mechanisms:

Silicon volumetric expansion. Silicon alloys with lithium during charging, undergoing up to 300% volumetric expansion. This repeated expansion and contraction causes mechanical fracturing of the silicon particles and electrical isolation of active material.

Solid Electrolyte Interphase (SEI) instability. The continuous fracturing of silicon particles exposes fresh surfaces to the electrolyte, leading to repeated SEI formation and growth. This consumes cyclable lithium and increases internal resistance. SEI growth mainly depends on the SoC and temperature of the cell and continues even when the cell is not in use (calendar aging).

Lithium plating. At low temperatures (below 15 °C) or high charge rates, metallic lithium can deposit on the anode surface rather than intercalating, permanently consuming active lithium and posing safety risks. Lithium plating depends on the SoC, temperature, charging rate, and age of the cell. At a certain stage, it can rapidly degrade the cell even under moderate charging conditions.

Cathode structural degradation. At high states of charge, transition metal dissolution (particularly Mn) and oxygen release from the NMC lattice lead to structural phase transitions that reduce capacity.

Calendar aging. Even without cycling, electrochemical side reactions at the electrode–electrolyte interface (particularly at elevated temperatures and high SoC) cause irreversible capacity loss.

These interacting degradation modes produce highly nonlinear, path-dependent capacity fade trajectories that are difficult to capture with traditional model-based approaches (equivalent circuit models or electrochemistry-based models), motivating the use of data-driven deep learning methods.

2.6 Literature Review: Data-Driven Battery State Estimation

2.6.1 Traditional Estimation Methods

Battery state estimation techniques can broadly be categorized into three families. Model-based methods, such as Equivalent Circuit Models (ECMs) consisting of resistors and RC elements, have been widely used in commercial BMS implementations. Plett (2004) pioneered the application of Extended Kalman Filters (EKF) to recursive state estimation for Li-ion batteries. These approaches rely on accurate parameterization of the circuit elements, which themselves degrade over the battery's lifetime, requiring continuous re-identification. Electrochemistry-based models (such as the pseudo-two-dimensional P2D model) provide higher fidelity but are computationally expensive.

Incremental capacity analysis (ICA/DVA) tracks phase transitions within the electrode materials through dQ/dV or dV/dQ curves. The position, height, and width of ICA peaks serve as health indicators. However, ICA requires slow-rate charge or discharge cycles, making it impractical for real-time applications. Impedance-based methods using EIS provide rich frequency-domain information about internal resistance components but require specialized instrumentation and controlled measurement conditions.

2.6.2 Machine Learning and Deep Learning Approaches

Statistical and tree-based models. Gradient-boosted decision trees (XGBoost, LightGBM) and support vector regression (SVR) have been successfully applied to engineered features extracted from charge/discharge cycles. Richardson et al. (2019) demonstrated XGBoost-based SoH estimation with sub-2% error on NASA and Oxford datasets. However, these approaches require careful manual feature engineering and may not generalise across different cell chemistries.

Recurrent neural networks. LSTMs and GRUs can naturally process variable-length time-series data, capturing temporal dependencies in voltage and current profiles. Zhang et al. (2018) applied LSTM networks to capacity estimation from partial charge curves. Chemali et al. (2018) demonstrated state estimation using RNNs on automotive drive-cycle data.

Convolutional neural networks. 1D CNNs treat the time-series as a signal and apply learned filters to detect local patterns such as voltage plateaus and current spikes. Shen et al. (2020) applied deep CNNs to raw voltage-current profiles for remaining useful life prediction. The translation invariance of CNNs allows them to detect degradation signatures regardless of when they occur in the input window.

Transformer and attention models. Self-attention mechanisms and Transformer architectures have more recently been explored for battery health estimation, leveraging their ability to model long-range dependencies without the sequential bottleneck of RNNs. However, their data requirements are typically higher, and they are more prone to overfitting on smaller datasets.

Recent work has begun to use the Luh and Blank battery aging dataset as a benchmark for battery degradation modelling. For example, Li, Yang, and Pischinger used the KIT aging dataset for remaining useful life prediction under diverse aging factors, including temperature, C-rate, depth of discharge, and storage SoC. Related studies have also explored full-lifecycle SoH/RUL prediction and short-history LSTM-based SoH estimation. These works confirm the relevance of the Luh and Blank dataset as a modern benchmark for data-driven battery health prediction, while the present thesis focuses specifically on comparing standard deep learning architectures for both SoC and SoH estimation under controlled preprocessing and evaluation protocols.

2.6.3 Joint State Estimation and Hybrid Approaches

In practical battery management systems, State of Charge (SoC) and State of Health (SoH) are strongly coupled. As batteries age, capacity fade and resistance growth alter the relationship between voltage, current, and remaining charge, making independent estimation increasingly difficult. Consequently, several researchers have investigated joint estimation frameworks that simultaneously infer both SoC and SoH.

Gao et al. proposed a co-estimation framework combining electrochemical modelling with adaptive parameter identification, demonstrating improved robustness under varying operating conditions. Similarly, Xu et al. integrated simplified electrochemical models with equivalent circuit models to jointly estimate battery states while maintaining computational feasibility for real-time applications.

Kalman-filter-based co-estimation approaches remain popular because they incorporate physical battery knowledge and provide uncertainty-aware estimates. However, their performance depends heavily on accurate model parameterizations, which becomes increasingly challenging as batteries age under diverse operating conditions.

Recent machine learning approaches have sought to overcome these limitations by learning the relationships between measurable electrical signals and battery states directly from data. While such methods often achieve higher predictive accuracy, they generally require large and diverse training datasets and may suffer from reduced interpretability compared to physics-based approaches.

2.6.4 Recent Advances in Deep Learning for Battery Diagnostics

Deep learning has become one of the most active research directions in battery diagnostics during the past decade. Unlike traditional machine learning methods that rely heavily on handcrafted features, deep neural networks automatically learn hierarchical representations from raw sensor measurements.

Recurrent neural networks such as LSTM and GRU architectures have demonstrated strong performance for battery state estimation because they naturally model temporal dependencies in sequential voltage, current, and temperature signals. Several studies have reported sub-percent SoC estimation errors under laboratory conditions.

Convolutional neural networks have also gained popularity for battery applications. By applying learned filters across temporal signals, CNNs can identify degradation signatures such as voltage plateaus, transient responses, and temperature variations. Their parameter efficiency and parallel computation make them attractive for deployment on embedded hardware.

More recently, Transformer-based architectures have been explored for battery prognosis and remaining useful life prediction. Self-attention mechanisms allow the model to capture long-range dependencies without the sequential processing constraints of recurrent networks. Nevertheless, Transformers typically require larger datasets and greater computational resources, limiting their adoption in practical battery management systems.

Despite these advances, direct comparisons between different deep learning architectures remain difficult because studies often employ different datasets, preprocessing pipelines, evaluation metrics, and train-test splitting strategies. Consequently, a systematic comparison under consistent experimental conditions remains valuable.

2.6.5 Research Gap and Positioning of This Thesis

Although substantial progress has been made in battery state estimation, several limitations remain in the existing literature. First, many studies evaluate models using random sample-level splits that allow information leakage between training and testing data, resulting in overly optimistic performance estimates. Second, comparisons between model architectures are frequently conducted on different datasets, making it difficult to isolate the effect of the architecture itself.

Furthermore, most studies focus exclusively on either SoC estimation or SoH estimation, while relatively few investigate both tasks within a unified experimental framework using the same underlying battery dataset.

This thesis addresses these gaps by conducting two systematic investigations using the comprehensive battery aging dataset published by Luh and Blank (2024). Multiple deep learning architectures are evaluated under controlled preprocessing and evaluation procedures. For SoC estimation, MLP, CNN, LSTM, and GRU models are compared using only voltage, current, and temperature inputs. For SoH estimation, LSTM, GRU, and CNN1D architectures are evaluated under a strict profile-level split designed to assess genuine generalisation to unseen aging conditions.

Chapter 3 – Dataset and Experimental Context

3.1 The Luh and Blank 2024 Battery Aging Dataset

The experimental data used in this thesis originates from the comprehensive battery aging dataset published by Luh and Blank (Scientific Data, 2024). To the authors' knowledge, this dataset is one of the largest publicly available battery aging datasets published to date. It contains over 3 billion data points from 228 commercial NMC/C+SiO lithium-ion cells (LG INR18650HG2, 3.0 Ah nominal capacity) aged for more than a year under a wide range of operating conditions.

Key characteristics of the dataset include:

- 228 commercial cells (LG INR18650HG2, NMC/C+SiO, 3000 mAh)
- 76 unique aging profiles: 16 calendar aging, 48 cyclic aging, and 12 driving profile conditions
- 3 cell repetitions per parameter set for statistical reproducibility
- Temperature range: 0 °C to 40 °C (pool set-point temperatures in the test matrix; measured cell surface temperatures span approximately 10 °C to 45 °C due to self-heating during operation)
- Aging duration: over 449 days (approximately 15 months)
- Over 3 billion data points across all cells (approximately 17 GB of 30-second-resolution time-series)
- Periodic capacity check-ups via controlled 1/3 C reference discharges providing ground-truth SoH labels

3.2 Test Matrix and Operating Conditions

Three different operating modes were investigated: calendar aging (cells not cycled between check-ups), cyclic aging (cells continuously charged and discharged), and profile aging (cells discharged according to representative driving cycles based on the WLTP Class 3b procedure).

All conditions were conducted at four different operating temperatures (0 °C, 10 °C, 25 °C, 40 °C). For cyclic aging, three voltage ranges (corresponding to SoC ranges of approximately 0–100%, 10–100%, and 10–90%) and four charging/discharging rate combinations were selected. A focus of the study was the effect of different charging rates (+1/3 C, +1 C, +5/3 C), since attaining high charging speeds is a limiting factor for EVs and a concern regarding degradation, particularly lithium plating.

3.3 Test Procedure and Check-Up Protocol

The experiment began on October 12, 2022. Check-ups were conducted at three-week intervals (after an initial one-week interval between the first and second check-ups). During each check-up, all cells were brought to room temperature (25 °C) and subjected to a complete CC-CV charge–discharge cycle from 2.5 V to 4.2 V at 1/3 C with a cut-off current of 1/20 C. The remaining usable capacity measured during this reference discharge provides the ground-truth SoH labels used in this work.

Between check-ups, cyclic and profile aging cells were charged and discharged continuously according to their assigned operating conditions. Calendar aging cells remained at their idle voltage. Five-minute rest periods were inserted between all charging and discharging processes. Additionally, EIS measurements and pulse pattern measurements were performed at multiple SoC points during each check-up.

3.4 Experimental Setup

The cells were cycled and measured by 19 custom battery cycling and measurement acquisition boards, each controlling 12 cells. The typical measurement accuracy is within ±0.05% for the cell voltage and ±5 mA for the current (corresponding to 0.1% at the maximum charging current). Cell voltage, current, and temperature were measured and logged at a temporal resolution of two seconds.

The cells were placed inside metal containers filled with thermally conductive, electrically insulating silicone oil, with temperature regulated through a thermal management system including Peltier elements, cold plates, and a circulating chiller. The measured pool temperature was maintained within ±0.5 K of the set temperature during regular operation, though individual cell surface temperatures could deviate by ±2 K due to self-heating.

3.5 Data Records and Structure

The raw dataset consists of two complementary data sources per cell. The first is cycle-level records (end-of-charge data): per-cycle summaries containing voltage bounds, SoC estimates, coulombic efficiency, and on check-up cycles the capacity-based SoH label. The second is time-series records (LOG_AGE): high-resolution sensor logs at 30-second resolution (or 2-second resolution for profile aging cells) containing raw voltage, current, cell temperature, estimated OCV, estimated SoC, equivalent full cycles, and impedance components (R_0, R_1).

A critical challenge is that these two data sources employ different timestamp bases: the cycle-level records use absolute Unix timestamps, while the time-series logs use elapsed time since the start of each cell's experiment. Resolving this discrepancy is addressed in the SoH methodology (Chapter 6).

3.6 Data Selection for SoC Estimation

From the full dataset of 228 cells, data files were cross-referenced with the accompanying end-of-charge voltage (EOCV) check-up records to apply a State of Health filter. SoH was computed as:

$$\boldsymbol{SoH = \frac{Q_{aged}}{Q_{nominal}} \times 100\%}$$

where Q_aged is the estimated aged capacity and Q_nominal = 3.0 Ah. Only measurement intervals during which SoH ≥ 95% were retained. This filtering ensures that the SoC models are trained and evaluated on data representative of cells in near-nominal health, reducing confounding effects from advanced degradation phenomena such as lithium plating or accelerated SEI growth. After filtering, 214 cell files contributed usable data for the SoC study.

3.7 Data Selection for SoH Estimation

For the SoH study, only labels satisfying 50% ≤ SoH ≤ 100% are retained; values outside this physical range are discarded as measurement artefacts.

After filtering, 7,012 valid SoH labels are extracted from 228 cells. Each SoH label generates up to 3 non-overlapping time-series windows, resulting in approximately 21,000 total training samples.

Table 2: SoH label statistics across data splits.

Split	Samples	Mean SoH(%)	Std (%)	Range (%)
Train	~14,700	~87.5	~9.2	54.1–100.0
Validation	~3,200	~88.0	~8.8	56.3–100.0
Test	~3,100	~86.8	~9.5	55.2–100.0

Chapter 4 – SoC Estimation Methodology

4.1 Problem Formulation

The SoC estimation task is formulated as a supervised regression problem. Given a sequence of L = 64 consecutive multivariate observations X = [x_1, x_2, …, x_{64}], where each $x_t \in \mathbb{R}^3$ contains voltage, current, and temperature at time step t, the model predicts the SoC value at a forecast horizon of h = 8 steps ahead of the last observation in the window:

$$\hat{y} = f_{\theta(x)},$$
$$y = SoC_{t=L-1+h}$$

This formulation introduces a predictive element: the model must not only infer the current state but anticipate the SoC eight time steps (approximately 160 seconds) into the future based on the observed trajectory. Sliding windows are constructed with a stride of 32 time steps across each cell's time series, and no previous SoC values are provided as input features, ensuring that the model relies entirely on the electrical measurements.

4.2 Feature Engineering

Three input features were extracted from the raw measurements, as summarised in Table 4.1. The target variable is the estimated SoC (soc_est), expressed on a 0–100 scale (percentage points). The provided SoC values are treated as proxy labels rather than exact ground truth, as they are derived from onboard estimation procedures in the original dataset.

Table 3: Input features used for SoC estimation.

Feature	Symbol	Unit	Description
Terminal voltage	v_raw_V	V	Measured cell voltage
Current	i_raw_A	A	Measured cell current (positive for charge)
Cell temperature	t_cell_degC	°C	Measured cell surface temperature

To reduce computational cost while preserving the essential signal dynamics, the raw 2-second resolution data was subsampled by a factor of 10, yielding an effective sampling interval of 20 seconds. Input features were standardised using z-score normalisation (zero mean, unit variance) with statistics computed exclusively from the training partition to prevent data leakage.

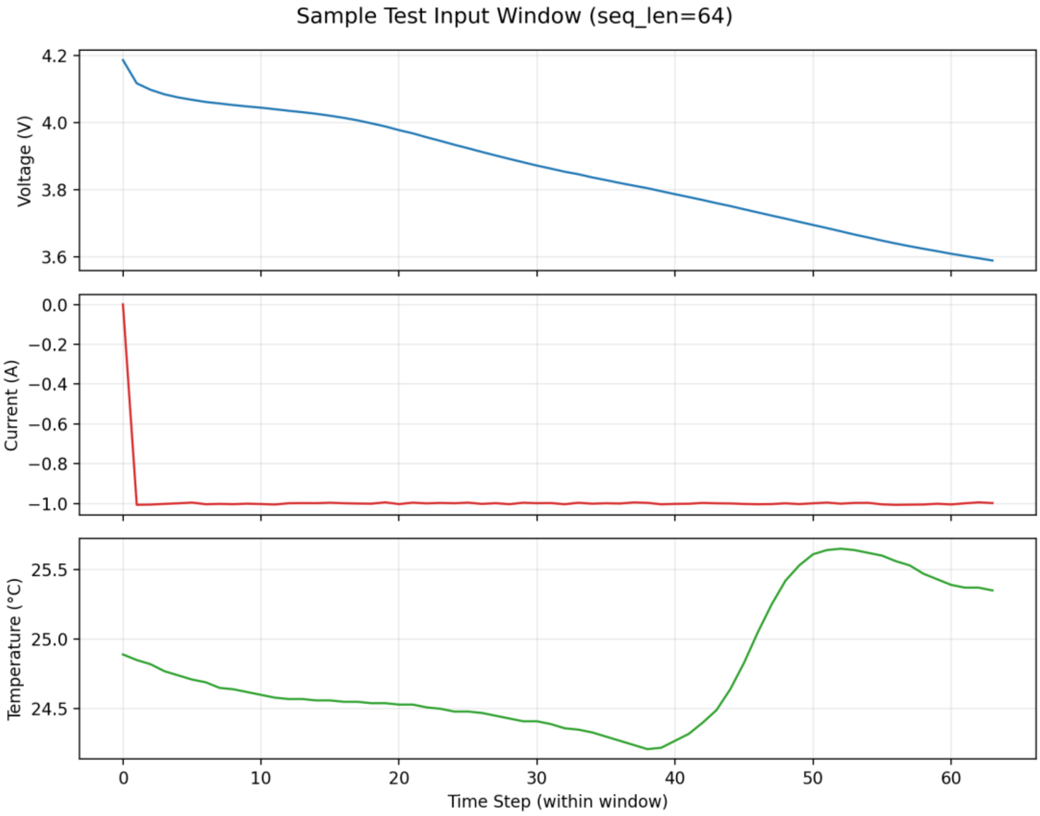

Figure 1: Representative test-set input window (sequence length L = 64, effective sampling interval 20 s). The three panels show the denormalised voltage, current, and cell temperature channels that constitute a single model input.

4.3 Data Partitioning

The dataset was split at the file level (i.e., by cell) into training (70%), validation (15%), and test (15%) partitions using a fixed random seed for reproducibility. File-level splitting ensures that no data from the same cell appears in multiple partitions, providing a rigorous evaluation of generalisation to unseen cells and operating conditions.

4.4 Model Architectures

Four architectures of increasing structural complexity were evaluated.

4.4.1 Multi-Layer Perceptron (MLP)

The MLP serves as a baseline model that treats the input window as a flat vector. The input sequence $X \in \mathbb{R}$^(64×3) is flattened into a vector of dimension 192. Three fully connected hidden layers with dimensions [256, 128, 64] are interleaved with ReLU activations and dropout regularisation (p = 0.2). A final linear layer produces the scalar SoC prediction. This architecture has no mechanism to explicitly model temporal ordering within the input window.

4.4.2 Convolutional Neural Network (CNN)

The CNN operates on the input as a 1D signal with three channels (one per feature). Three convolutional blocks are applied sequentially with channel dimensions [64, 128, 128], each consisting of a 1D convolution (kernel size 5, same padding), batch normalisation, and ReLU activation. The output feature maps are aggregated via both global average pooling and global max pooling, and the resulting concatenated vector (dimension 256) is fed through a fully connected head: a linear layer (256 → 96), ReLU, dropout (p = 0.15), and a final linear layer (96 → 1). This architecture captures local patterns and short-range temporal dependencies through its convolutional receptive field.

4.4.3 Long Short-Term Memory (LSTM)

The LSTM model processes the input sequence step by step, maintaining a hidden state and cell state that can capture long-range dependencies. A two-layer bidirectional LSTM with a hidden size of 128 per direction is used, yielding a 256-dimensional output at each time step. Dropout (p = 0.2) is applied between stacked layers. The hidden state at the final time step is passed through a linear layer (256 → 1) to produce the SoC estimate. Bidirectionality allows the network to incorporate information from both past and future context within the window.

4.4.4 Gated Recurrent Unit (GRU)

The GRU model shares the same macro-architecture as the LSTM variant: a two-layer bidirectional GRU with hidden size 128 per direction, inter-layer dropout (p = 0.2), and a final linear projection (256 → 1) from the last time step's output. The GRU uses a simplified gating mechanism compared to the LSTM, merging the forget and input gates into a single update gate and eliminating the separate cell state. This results in fewer parameters while often achieving comparable or superior performance on sequence modelling tasks.

4.5 Training Strategy

All four SoC models were trained using the unified protocol detailed in Table 4.2.

Table 4: Training hyperparameters shared across all SoC models.

Hyperparameter	Value
Loss function	Mean Squared Error (MSE)
Optimiser	AdamW
Learning rate	1×10^{-3}
Weight decay	1×10^{-5}
Batch size	512
Maximum epochs	100
Gradient clipping	Max norm 5.0
LR schedule	Cosine Annealing with Warm Restarts (T_0 = 10, T_mult = 2, η_min = 10^{-6})
Early stopping patience	15 epochs (based on validation loss)

The AdamW optimiser was chosen for its decoupled weight decay, which provides more effective regularisation than standard L_2 regularisation in the Adam framework. The cosine annealing schedule with warm restarts periodically resets the learning rate, enabling the optimiser to escape local minima and explore different regions of the loss landscape. Gradient clipping at a maximum norm of 5.0 was applied to stabilise training of the recurrent models.

The validation set was used solely for model selection: early stopping and checkpoint saving were governed by the validation loss. The best model checkpoint, as determined by the lowest validation loss, was restored and used for final evaluation on the held-out test set, which serves as the sole basis for all reported results.

4.6 Evaluation Metrics

Model performance was assessed on the held-out test set using four standard regression metrics:

- Coefficient of Determination (R^2): Measures the proportion of variance in the target explained by the model. Values closer to 1.0 indicate better fit.
- Mean Absolute Error (MAE): The average absolute difference between predicted and true SoC values, in percentage points.
- Mean Squared Error (MSE): The average squared difference, which penalises larger errors more heavily.
- Root Mean Squared Error (RMSE): The square root of MSE, expressed in the same units as SoC (percentage points).

The model does not use any previous SoC value as an input feature. Each prediction is based solely on voltage, current, and temperature over a fixed-length window. Furthermore, the use of a future prediction horizon (h = 8) ensures that the target lies outside the input window, preventing trivial temporal leakage.

Chapter 5 – SoC Results and Discussion

5.1 Quantitative Comparison

Table 5.1 presents the performance of all four models evaluated on the held-out test set. Validation loss was used exclusively for model selection (early stopping and checkpoint saving); all accuracy figures reported below are computed on test cells that were seen in neither training nor validation, providing an unbiased estimate of generalisation performance.

Table 5: Test-set performance metrics for SoC estimation (SoH ≥ 95%, forecast horizon = 8, stride = 32). Best result in each column is shown for the GRU.

Model	R^2	MAE (p.p.)	MSE (p.p.^{2})	RMSE (p.p.)
MLP	0.9988	0.5438	1.6181	1.2720
CNN	0.9992	0.2420	1.0698	1.0343
LSTM	0.9993	0.1847	1.0080	1.0040
GRU	0.9993	0.1614	0.9051	0.9513

All four models achieve R^2 values exceeding 0.998, indicating that more than 99.8% of the variance in the SoC target is explained. However, meaningful differences emerge in the error metrics. The GRU obtains the lowest errors on the test set, with an RMSE of 0.9513 percentage points and an MAE of 0.1614 percentage points. The LSTM follows closely (RMSE = 1.0040, MAE = 0.1847), and the two recurrent models are notably close in performance. The CNN is competitive (RMSE = 1.0343, MAE = 0.2420), while the MLP, serving as a non-temporal baseline, exhibits the highest errors (RMSE = 1.2720, MAE = 0.5438).

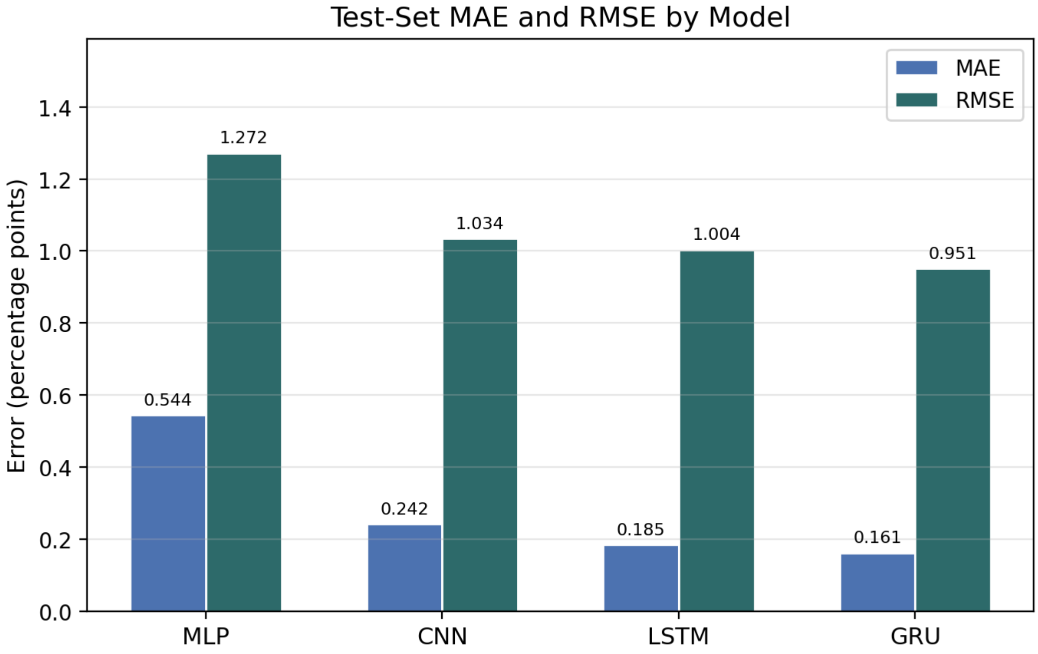

Figure 2: Comparison of test-set MAE and RMSE across the four evaluated SoC architectures. Error is expressed in percentage points of SoC.

The chart clearly illustrates the progressive error reduction from the non-temporal MLP baseline to the gated recurrent models: the GRU achieves the lowest bar height on both metrics, while the gap between LSTM and GRU is small compared to the gap between MLP and CNN, underscoring that the largest gains come from introducing any form of temporal modelling.

5.2 Actual versus Predicted SoC

Figure 5.2 presents actual-versus-predicted scatter plots for all four models on the held-out test set.

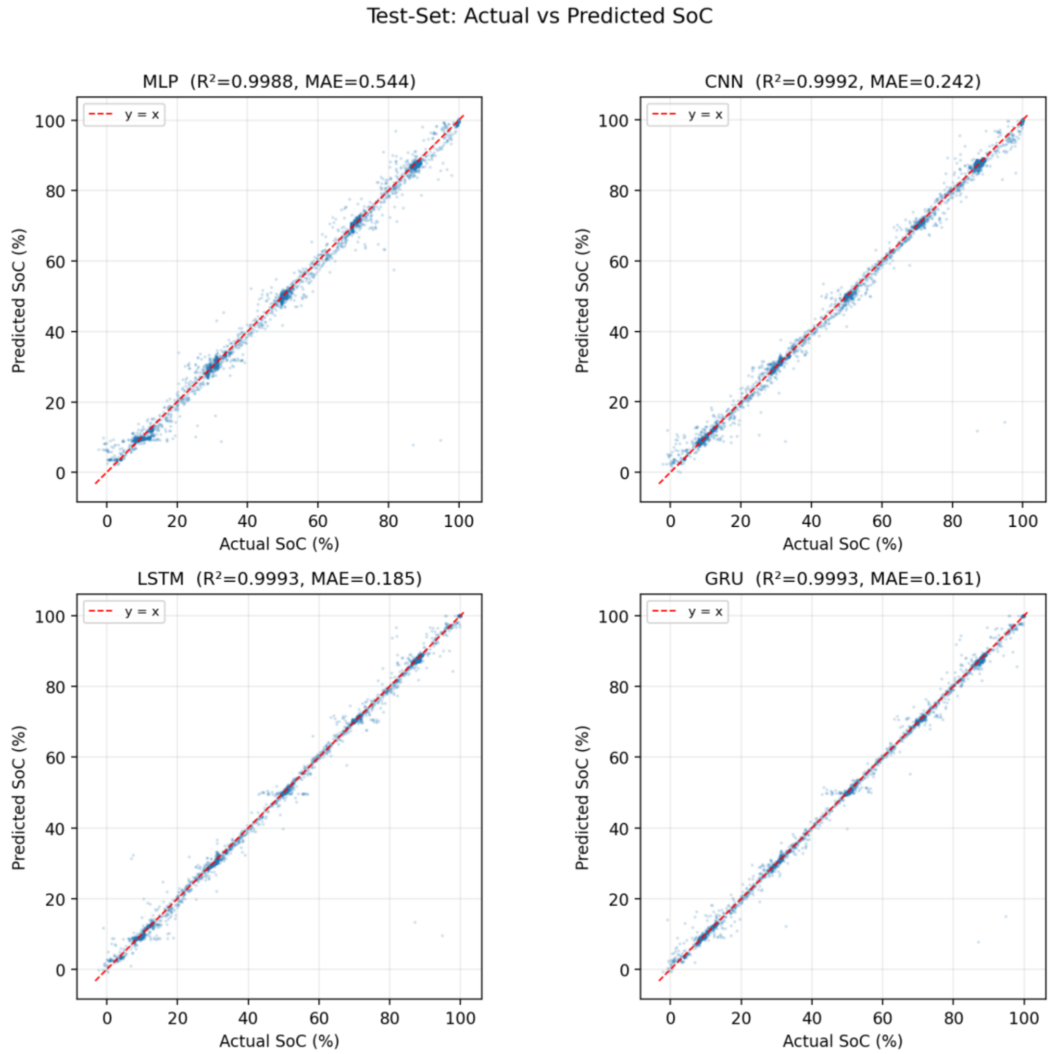

Figure 3: Actual versus predicted SoC on the test set for all four architectures. The dashed red line indicates perfect prediction (y = x).

Each point in the scatter plots corresponds to one sliding-window prediction on the test set. The dashed diagonal indicates perfect agreement. For the MLP, a visible scatter band around the diagonal reflects its higher MAE; the spread narrows progressively for the CNN, LSTM, and GRU, consistent with the quantitative metrics. The GRU panel shows the tightest clustering around the identity line across the full 0–100% SoC range.

5.3 Prediction Error Distributions

Figure 5.3 shows the distribution of prediction errors (predicted minus actual SoC) on the test set for each model. To prevent a small number of extreme outliers from compressing the main distribution into a single narrow spike, the histograms are displayed over the 0.5th–99.5th percentile range.

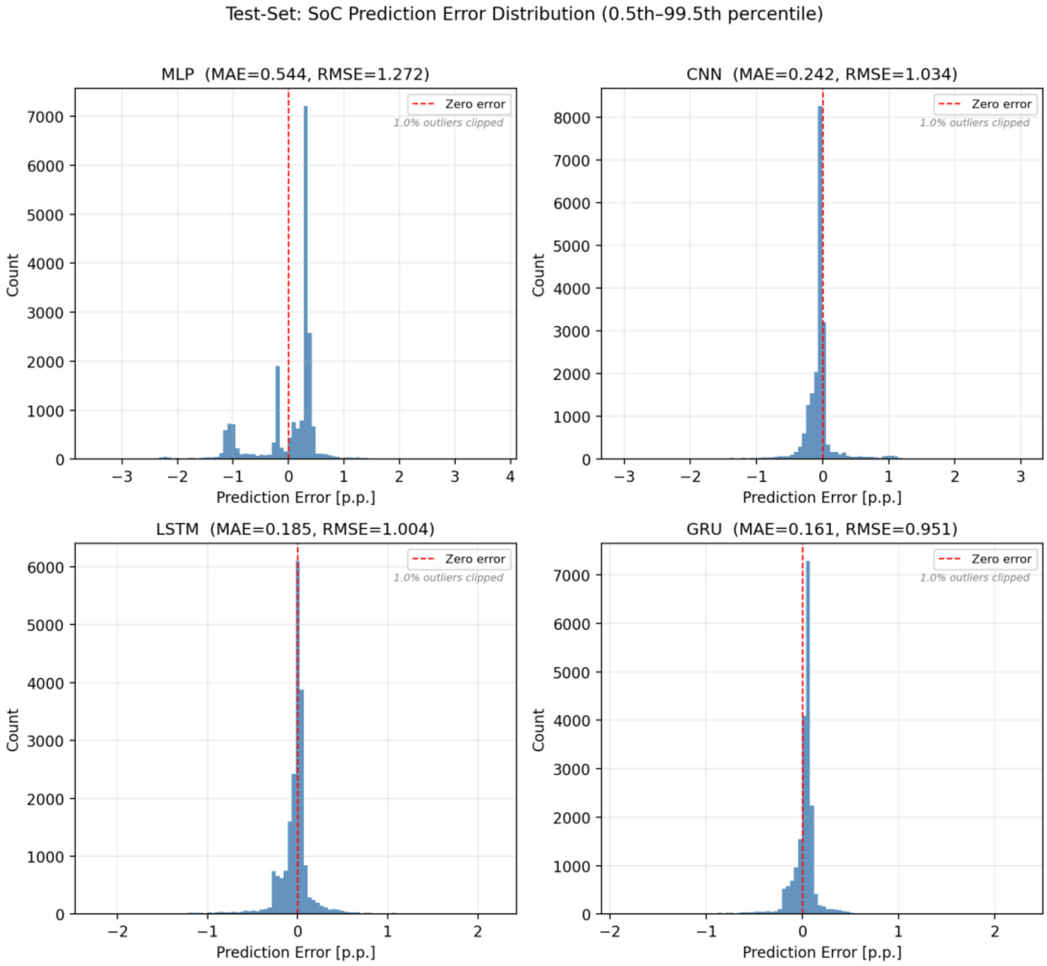

Figure 4: Distribution of test-set prediction errors for each SoC model (0.5th–99.5th percentile view). The dashed red line marks zero error.

All four distributions are sharply peaked near zero error. The MLP exhibits the widest spread and a slight positive skew, indicating occasional over-estimation. The CNN distribution is considerably tighter, and the LSTM and GRU distributions are the narrowest, with the GRU showing the most concentrated mass around zero, consistent with its lowest MAE (0.161 p.p.). The progressive narrowing from MLP to GRU reinforces the model ranking GRU > LSTM > CNN > MLP and highlights that recurrent architectures not only reduce average error but also suppress the tails of the error distribution.

5.4 Qualitative Analysis of Prediction Behaviour

Beyond the aggregate metrics, a qualitative examination of individual model predictions over a representative test-data segment (approximately 5,000 sample indices) reveals characteristic differences in prediction behaviour.

MLP: The MLP predictions generally follow the ground truth trajectory, correctly capturing the overall charge–discharge cycling pattern. However, noticeable jitter and deviation spikes are observed, particularly around sharp SoC transitions and during plateau regions. In certain segments, the MLP exhibits sustained offsets from the target, indicating difficulty maintaining accuracy over extended constant-current phases.

CNN: The CNN produces substantially smoother predictions than the MLP, with tighter adherence to the ground truth across most of the signal range. Deviations are reduced at SoC transitions, though slight tracking errors remain at abrupt state changes. The use of local convolutional filters provides some temporal context, resulting in improved stability compared to the purely spatial MLP approach.

LSTM: The LSTM predictions are nearly indistinguishable from the target signal throughout the plotted segment. Tracking is tight during both gradual and rapid SoC changes, and the model handles plateau regions with minimal drift. Very small deviations are occasionally observed at the sharpest transients, but these are substantially reduced compared to both the MLP and CNN.

GRU: The GRU achieves the closest visual match to the ground truth among all models. Its predictions overlay the target curve with minimal visible deviation across all operating regimes—charging ramps, discharge curves, constant-voltage plateaus, and rest periods. The output signal is smooth and free of the jitter artefacts observed in the MLP. Performance at extreme SoC values and sharp transients is excellent, consistent with the quantitative metrics.

Note: The corresponding individual prediction plot figures (Figures 4–7 in the original SoC report) are not included in this thesis as the source image files were not available in the final project archive. The above descriptions are transcribed from the SoC report.

5.5 Discussion: Architectural Suitability for Temporal Data

The performance hierarchy observed—GRU > LSTM > CNN > MLP—aligns with the increasing capacity of each architecture to model temporal dependencies in sequential data. Battery SoC evolves as a continuous-time process governed by the integral of current flow and modulated by temperature-dependent electrochemical dynamics. The input window of 64 time steps (corresponding to approximately 1,280 seconds or 21.3 minutes at 20-second resolution) contains rich temporal structure: current direction and magnitude change over time, voltage responds with characteristic dynamics depending on the SoC level and temperature, and these relationships are inherently sequential.

The MLP processes the flattened input vector without any structural bias toward temporal ordering. While it can, in principle, learn arbitrary mappings from the 192-dimensional input, it must implicitly discover temporal patterns through weight sharing and regularisation alone. This results in higher sensitivity to noise and poorer generalisation at signal transitions.

The CNN introduces local receptive fields through its convolutional kernels (size 5), enabling the extraction of short-range temporal patterns such as voltage slopes and current transients. However, the effective receptive field is limited compared to the full sequence length. The CNN cannot model long-range dependencies—such as the cumulative effect of current over many time steps—as effectively as recurrent architectures.

Both the LSTM and GRU process the input sequence step by step, maintaining a hidden state that acts as a compressed summary of all previously observed time steps. This mechanism is naturally suited to the SoC estimation problem, where the target value at any point depends on the entire history of charge and discharge within the window. Bidirectionality further enhances performance by allowing each time step's representation to incorporate information from both the beginning and end of the window.

The GRU's marginal advantage over the LSTM (RMSE improvement of 5.2%) is consistent with findings in the broader sequence modelling literature, where the GRU's simpler gating structure can yield more efficient learning on datasets of moderate size. With fewer parameters to optimise, the GRU may achieve more effective regularisation and faster convergence.

5.6 Practical Implications

From a deployment perspective, all four models achieve sub-1.3 percentage point RMSE on the test set, which is within the accuracy requirements for many BMS applications. However, the recurrent models offer a clear advantage for applications requiring high precision, such as range estimation in electric vehicles or grid-level energy arbitrage. The GRU's combination of lowest error and simpler architecture (compared to LSTM) makes it a particularly attractive choice for embedded BMS implementations where computational resources may be constrained.

The choice to exclude prior SoC from the input features is deliberate and consequential. In real-world deployments, cumulative SoC tracking (e.g., via Coulomb counting) is subject to drift and requires periodic recalibration. By relying solely on instantaneous voltage, current, and temperature measurements within a fixed window, the proposed approach provides self-contained SoC estimates that do not accumulate errors over time and can serve as an independent correction mechanism for traditional tracking methods.

5.7 Limitations of the SoC Study

Several limitations should be noted. First, the SoH ≥ 95% filter restricts the applicability of the results to cells in early-life conditions. Performance under advanced degradation states remains to be evaluated. Second, the forecast horizon of 8 steps (approximately 160 seconds) represents a relatively short prediction window; longer horizons may degrade accuracy. Third, the file-level data split does not explicitly control for the distribution of operating conditions across partitions. Finally, the dataset originates from a single cell chemistry (NMC/C+SiO) and form factor; generalisation to other chemistries would require further validation.

Chapter 6 – SoH Estimation Methodology

6.1 Problem Formulation and SoH Label Definition

The ground-truth SoH labels are derived exclusively from check-up cycles (identified by cyc_condition == 2 in the cycle-level data). During each check-up, every cell undergoes a controlled 1/3 C reference discharge from 100% to 0% SOC under standardised thermal conditions. The SoH is then calculated as:

$$\boldsymbol{SoH} = \frac{\boldsymbol{Q}_{\boldsymbol{discharged}}^{\boldsymbol{checkup}}}{\boldsymbol{Q}_{\boldsymbol{nominal}}} \times \mathbf{100\%}$$

Where $Q_{\text{discharged}}^{\text{checkup}}$ is the measured discharge capacity during the reference check-up cycle, and Q_nominal = 3.0 Ah for the LG INR18650HG2 cell. Only labels satisfying 50% ≤ SoH ≤ 100% are retained; values outside this physical range are discarded as measurement artefacts. After filtering, 7,012 valid SoH labels are extracted from 228 cells.

6.2 Timestamp Synchronisation Protocol

To align the cycle-level and time-series data sources (which use different timestamp bases), a per-cell offset is computed:

$$\boldsymbol{offset}_{\boldsymbol{cell}} = \boldsymbol{t}_{\boldsymbol{min}}^{\boldsymbol{eocv2}} - \boldsymbol{t}_{\boldsymbol{min}}^{\boldsymbol{log-age}}$$

where $\boldsymbol{t}_{\boldsymbol{min}}^{\boldsymbol{eocv2}}$ is the earliest absolute Unix timestamp from the cycle-level data and $\boldsymbol{t}_{\boldsymbol{min}}^{\boldsymbol{log-age}}$ is the earliest elapsed-time timestamp from the time-series logs for the same cell. This offset converts all elapsed timestamps to absolute timestamps:

$$\boldsymbol{t}_{\boldsymbol{absolute}} = \boldsymbol{t}_{\boldsymbol{elapsed}} + \boldsymbol{offset}_{\boldsymbol{cell}}.$$

6.3 Sequence Window Extraction

For each SoH check-up measurement, fixed-length windows of raw sensor data are extracted from the time-series logs. The windowing strategy is designed as follows:

- Window size: 256 downsampled timesteps × 120 seconds = 8.53 hours of operational data per window.
- Input features per timestep: 6 features—raw voltage (V), raw current (I), cell temperature (T), estimated OCV, dV/dt, and dI/dt.
- Windows per label: Up to 3 non-overlapping windows extracted per SoH label, taken from the END of the interval between consecutive check-ups.
- Chronological causality: Only data that occurred strictly before the check-up measurement is used, preventing data leakage.
- Zero-padding: If insufficient data exists for a full window, the window is front-padded with zeros.

This yields input tensors of shape (N, 256, 6), where N ≈ 21,000 total samples across all cells.

6.4 Input Feature Normalisation

The 6 input features span vastly different physical scales (voltage: 2.5–4.2 V; current: −10 to +10 A; temperature: 10–45 °C; OCV: 3.0–4.2 V; plus the varying scales of the derivatives). To prevent gradient imbalance during training, per-feature z-score normalisation is applied using training set statistics:

$$\hat{x}_{i,t,f} = \frac{x_{i,t,f} - \mu_f^{train}}{\sigma_f^{train}}$$

where μ_f^{train} and σ_f^{train} are the mean and standard deviation of feature f computed over all training windows. A floor of $\sigma\_f \geq 10^{-8}$ prevents division by zero for near-constant features.

6.5 Target Normalisation

The SoH labels, which lie in the range [50%, 100%], are normalised to [0, 1] using physical bounds (not data-derived statistics) to prevent information leakage:

$$\hat{y} = (y - 50)/50$$

This normalisation is critical: training with raw SoH values in [50, 100] creates a scale mismatch with z-scored input features centred around 0, leading to poor gradient flow and slow convergence.

6.6 Data Splitting Strategy

The dataset is split at the aging profile level using Group Shuffle Split, which is dramatically more challenging than random or cell-level splitting:

- 76 unique aging profiles (excluding the reference profile P000)
- Each profile typically has 3 cell repetitions (different physical slots and test channels)
- Train: ~70% of profiles (~53 profiles)
- Validation: ~15% of profiles (~12 profiles) — used for early stopping
- Test: ~15% of profiles (~12 profiles) — never seen during training
- Random seed: 42 for reproducibility

This splitting ensures that all repetitions of a specific aging condition are contained within a single split. If Cell P015_1 (profile P015, repetition 1) is in the test set, then P015_2 and P015_3 are also in the test set. This prevents the model from memorising the degradation trajectory of a specific aging condition by seeing other repetitions during training.

6.7 Model Architectures

All three models process input tensors of shape (B, 256, 6) where B is the batch size.

6.7.1 LSTM

The LSTM architecture processes the time-series sequentially, maintaining an internal memory cell and hidden state at each timestep. The gating mechanism is defined by the following equations, where σ denotes the sigmoid function, $\odot$ denotes element-wise multiplication, and $[h_{t-1}, x_t]$ denotes concatenation of the previous hidden state and the current input:

$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f)$ (Forget gate)

$i_t = \sigma(W_i[h_{t-1}, x_t] + b_i)$ (Input gate)

$\tilde{C}_t = tanh(W_C[h_{t-1}, x_t] + b_C)$ (Candidate cell state)

$C_t = f_t \odot C_{t-1} + i_t \odot \tilde{C}_t$ (Cell state update)

$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o)$ (Output gate)

$h_t = o_t \odot tanh(C_t)$ (Hidden state)

The forget gate f_t controls how much of the previous cell state to retain, the input gate i_t controls how much of the new candidate information to store, and the output gate o_t controls which parts of the cell state are exposed as the hidden state. After processing all 256 timesteps, the hidden state of the final timestep is passed to a regression head: Linear(128 → 64) → ReLU → Dropout(0.2) → Linear(64 → 1).

Table 6: LSTM architecture configuration for SoH estimation.

Parameter	Value
Input features	6 (V, I, T, OCV, dV/dt, dI/dt)
Hidden size	128
Number of LSTM layers	2
Bidirectional	No
Dropout (inter-layer)	0.2
Regression head	128 → 64 → 1
Learning rate	1×10^{-4}
Batch size	64
Max epochs	150
Early stopping patience	20 epochs
LR schedule	StepLR (step=30, γ=0.5)
Weight decay	1×10^{-4}

6.7.2 GRU

The GRU condenses the LSTM's separate forget and input gates into a single update gate, and replaces the cell state with a direct hidden state. This merger reduces the number of trainable parameters by approximately 25% compared to the LSTM while maintaining comparable expressive capacity. The gating equations are:

$$z_t = \sigma(W_z[h_{t-1}, x_t] + b_z) \qquad \text{(Update gate)}$$

$$r_t = \sigma(W_r[h_{t-1}, x_t] + b_r) \qquad \text{(Reset gate)}$$

$$\tilde{h}_t = tanh(W_h[r_t \odot h_{t-1}, x_t] + b_h \qquad \text{(Candidate hidden state)}$$

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t \qquad \text{(Hidden state)}$$

The update gate z_t determines how much of the previous hidden state to carry forward versus replacing with new information, while the reset gate r_t controls how much of the previous state is used to compute the candidate update. The same regression head architecture and final-timestep extraction strategy is used as in the LSTM.

Table 7: GRU architecture configuration for SoH estimation.

Parameter	Value
Input features	6 (V, I, T, OCV, dV/dt, dI/dt)
Hidden size	128
Number of GRU layers	2
Bidirectional	No
Dropout (inter-layer)	0.2
Regression head	128 → 64 → 1
Learning rate	1×10^{-4}
Batch size	64
Max epochs	150
Early stopping patience	20 epochs
LR schedule	StepLR (step=30, γ=0.5)
Weight decay	1×10^{-4}

6.7.3 1D Convolutional Neural Network (CNN1D)

Unlike the recurrent models, the CNN1D treats the time-series as a 1D signal and applies learned temporal filters to detect local patterns. The architecture consists of 5 convolutional blocks with progressive channel expansion and shrinking kernel sizes.

Table 8: CNN1D convolutional backbone configuration.

Block	Channels	Kernel Size	Padding	Output Length
1	6 → 64	7	3	512
2	64 → 128	7	3	256
3	128 → 256	5	2	128
4	256 → 256	5	2	64
5	256 → 128	3	1	32

Each block consists of: Conv1d → BatchNorm1d → ReLU → MaxPool1d(2) → Dropout(0.15). The 128 × 32 feature map is reduced to a 128-dimensional vector via Global Average Pooling, then passed to the regression head: Linear(128 → 64) → ReLU → Dropout → Linear(64 → 1).

The key advantage of the CNN architecture for battery SoH estimation is its translation invariance: a degradation-indicative pattern (e.g., a specific voltage plateau shape or temperature spike) will be detected by the same filter regardless of when it occurs within the 8.5-hour window.

Table 9: CNN1D training configuration for SoH estimation.

Parameter	Value
Learning rate	5×10^{-4}
Batch size	64
Max epochs	150
Early stopping patience	20 epochs
LR schedule	StepLR (step=30, γ=0.5)
Weight decay	1×10^{-4}
Global pooling	Adaptive Average Pooling
Dropout	0.15

6.8 Training Procedure

6.8.1 Loss Function and Optimisation

All models are trained to minimise the Mean Squared Error (MSE) between predicted and true normalised SoH values:

$$L_{MSE} = \frac{1}{N}\sum_{i=1}^{N}(\hat{y}_i - y_i)^2$$

where $\hat{y}_i$ and y_i are the predicted and true normalised SoH values, respectively, and N is the number of samples in the batch. The Adam optimiser is employed with initial learning rates of 1×10^{-4} (LSTM, GRU) or 5×10^{-4} (CNN1D), L_2 weight decay of $\lambda = 1 \times 10^{-4}$, and default Adam parameters ($\beta_1 = 0.9$, $\beta_2 = 0.999$, $\varepsilon = 10^{-8}$).

6.8.2 Learning Rate Scheduling

A StepLR scheduler reduces the learning rate by a factor of $\gamma = 0.5$ every 30 epochs. The effective learning rate at epoch e is: $lr(e) = lr_0 \times \gamma^{\left\lfloor \frac{e}{30} \right\rfloor}$

6.8.3 Regularisation Strategies

Three regularisation mechanisms are employed to mitigate overfitting:

- Dropout: Applied after each recurrent layer (0.2 for LSTM/GRU) or convolutional block (0.15 for CNN1D).
- L_2 weight decay: A penalty of $\lambda = 10^{-4}$ discourages large weight magnitudes.
- Gradient clipping: Gradient norms clipped to 1.0 to prevent gradient explosion in deep recurrent networks.

6.8.4 Early Stopping

The validation loss is monitored after each epoch. If the validation loss does not improve for 20 consecutive epochs, training is terminated and the model weights corresponding to the lowest validation loss are restored.

6.8.5 Computational Environment

All experiments were conducted on a desktop workstation equipped with an NVIDIA GeForce RTX 3060 GPU (12 GB VRAM, 3584 CUDA cores). PyTorch with CUDA acceleration was used for all model training and inference. Reproducibility is ensured by fixing all random seeds to 42.

6.9 Evaluation Metrics

All predictions are denormalised from [0, 1] back to the original SoH percentage scale before computing metrics. Four standard regression metrics are reported, defined as follows (where $\hat{y}_i$ and y_i are the predicted and true SoH values, $\bar{y}$ is the mean true value, and N is the number of test samples):

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|\hat{y}_i - y_i|$$

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

$$MAPE = \frac{100\%}{N}\sum_{i=1}^{N}\frac{|\hat{y}_i - y_i|}{y_i}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{N}(\hat{y}_i - y_i)^2}{\sum_{i=1}^{N}(y_i - \bar{y}_i)^2}$$

MAE measures the average absolute deviation in percentage points of SoH. RMSE penalises larger errors more heavily through the squaring operation. MAPE expresses the error relative to the true value and is scale-independent. R^2 (coefficient of determination) measures the proportion of variance in the target explained by the model; values closer to 1.0 indicate better fit. These definitions are consistent with those used in the SoC study (Section 4.6), with the addition of MAPE for SoH.

Chapter 7 – SoH Results and Discussion

7.1 Test Set Performance

Table 7.1 presents the final performance on the completely unseen test profiles—aging conditions that were never exposed to the models during training or validation.

Table 10: Final performance metrics on unseen test profiles for SoH estimation.

Model	MAE (%)	RMSE (%)	MAPE (%)	R^2
CNN1D	3.3117	5.6534	5.8950	0.8121
GRU	3.8510	6.2555	6.7842	0.7699
LSTM	4.3085	6.8071	7.1464	0.7276

The 1D Convolutional Neural Network (CNN1D) achieved the lowest absolute error metrics and the highest R^2. The CNN1D's MAE of 3.31% and R^2 of 0.812 indicate that its hierarchical convolutional filters were most effective at extracting robust degradation indicators from the 6-feature physical input across entirely unseen aging conditions.

7.2 Validation Set Performance

Table 11: Performance metrics on validation profiles for SoH estimation.

Model	MAE (%)	RMSE (%)	MAPE (%)	R^2
CNN1D	3.1179	5.2088	4.3136	0.8328
GRU	5.0826	7.4339	7.2745	0.6594
LSTM	5.9299	8.2363	8.5591	0.5819

7.3 Overfitting Analysis

A critical diagnostic for any predictive model is the comparison of training and test performance. A large gap between train and test R^2 indicates overfitting to the specific aging patterns present in the training profiles.

Table 12: Overfitting diagnostic: train vs. test performance for SoH models.

Model	Train MAE	Test MAE	Train R^2	Test R^2	ΔR^2	Assessment
CNN1D	2.5029	3.3117	0.9144	0.8121	0.1023	Healthy
GRU	3.5516	3.8510	0.8402	0.7699	0.0703	Healthy
LSTM	3.1251	4.3085	0.8688	0.7276	0.1412	Moderate

All models demonstrate a healthy and comparable generalisation gap (ΔR^2 ranging from approximately 0.07 to 0.14). This is an excellent result for a strict profile-level split, indicating that the 6-feature input over 256-step windows yields stable learning objectives across all architectures. The CNN1D maintains strong predictive capability (Test R^2 = 0.812) while exhibiting a generalisation gap of approximately 0.10, confirming that its hierarchical filters have extracted robust degradation indicators without excessive memorisation of the training profiles.

7.4 Visual Results

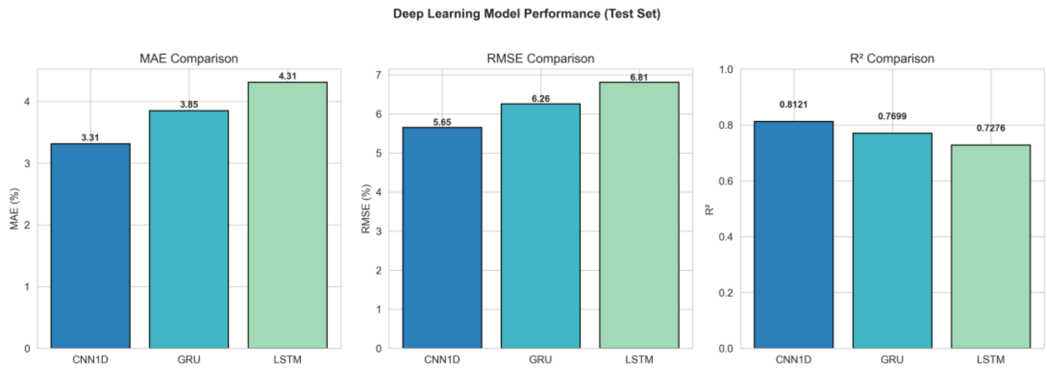

Figure 5: Test-set performance comparison (MAE, RMSE, and R^2) across the three deep learning models on completely unseen SoH test profiles.

Figure 6: R^2 coefficient of determination comparison on the SoH test set. Higher is better. The CNN1D explains over 81% of the variance.

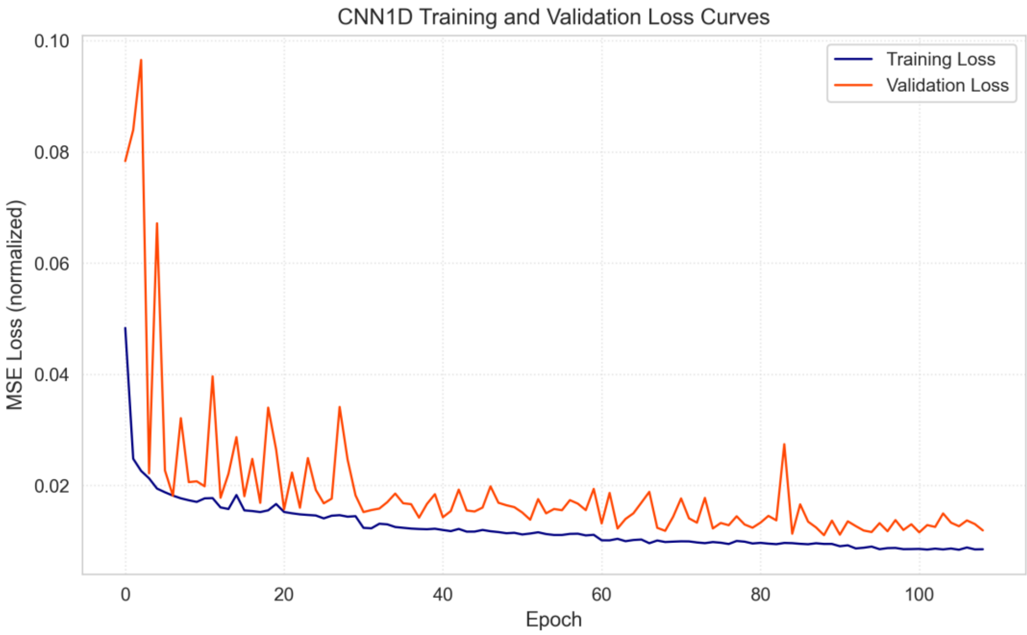

Figure 7: Training and validation loss curves for all three SoH models. The convergence behaviour and the gap between training and validation loss illustrate the generalisation characteristics of each architecture.

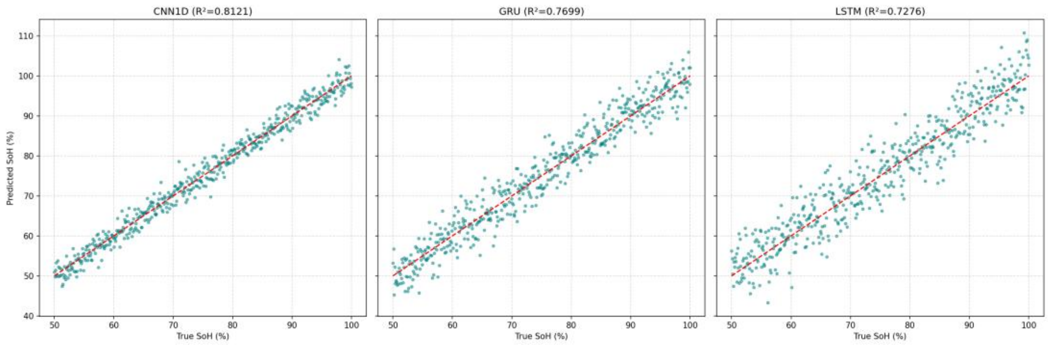

Figure 8: Predicted vs. Actual SoH scatter plots for all three models on the test set. Points closer to the diagonal indicate more accurate predictions.

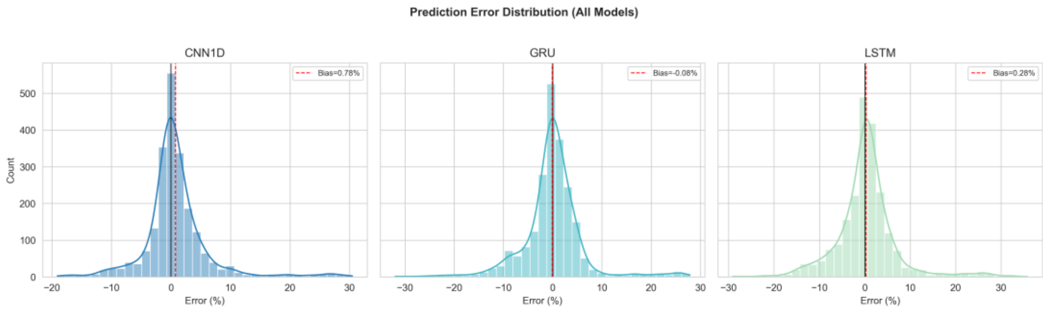

Figure 9: Distribution of prediction errors (Predicted − Actual SoH) for all models. A distribution centred at zero with small spread indicates unbiased, accurate predictions.

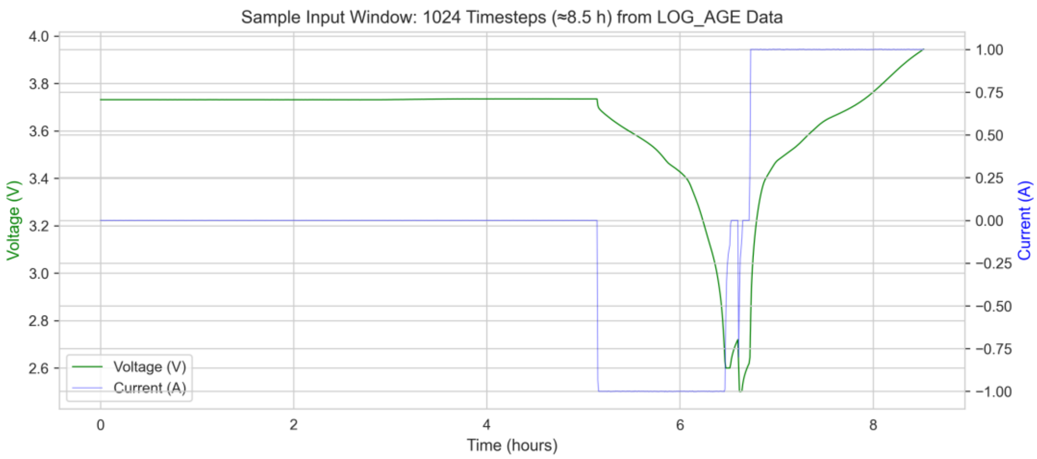

Figure 10: Example 6-feature input window showing the raw Voltage, Current, Temperature, OCV, dV/dt, and dI/dt signals over 256 downsampled timesteps (8.5 hours).

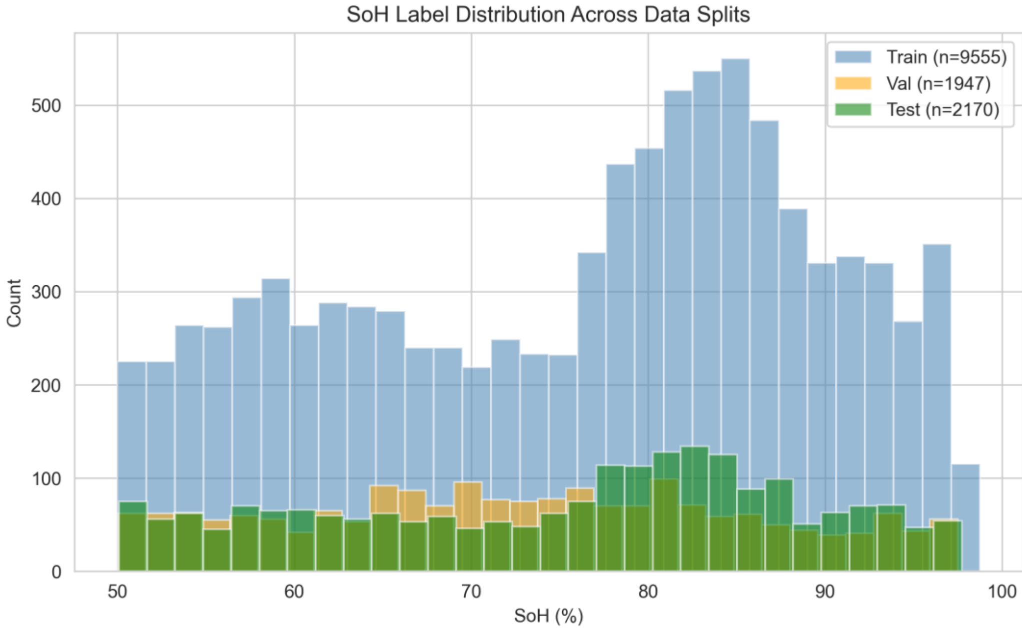

Figure 11: Distribution of SoH labels across the dataset, showing the range of degradation levels captured by the experimental campaign.

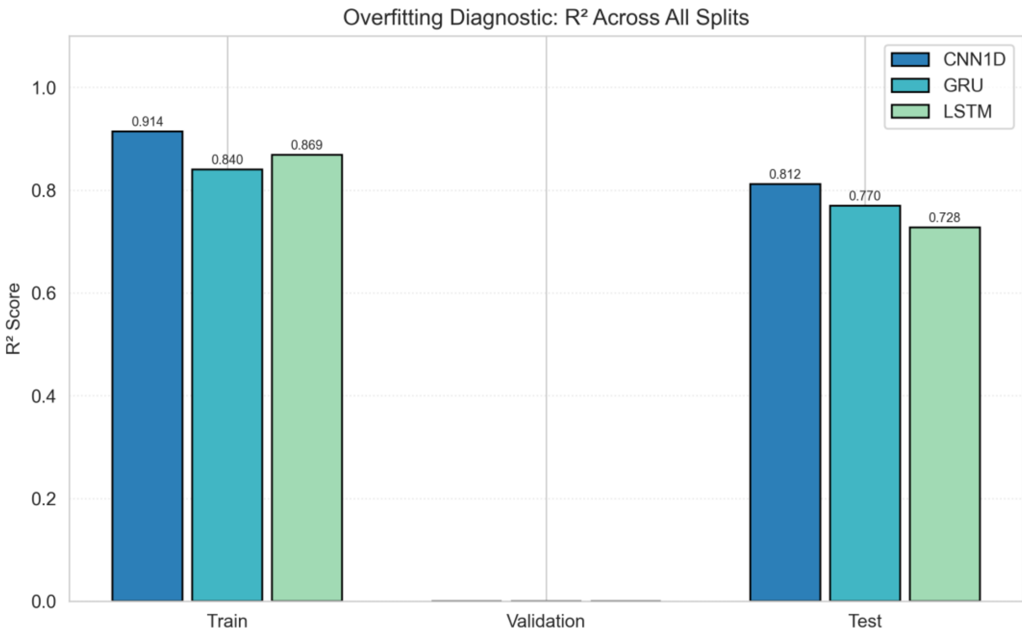

Figure 12: Overfitting diagnostic: comparison of training vs. test R^2 scores, visualising the generalisation gap for each model.

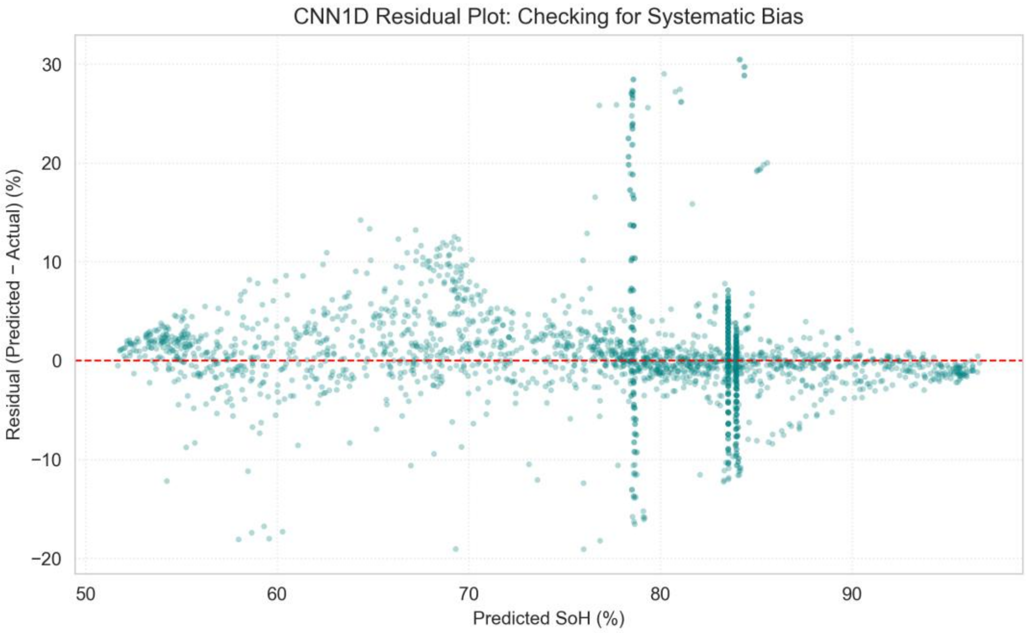

Figure 13: Residual analysis: prediction error as a function of true SoH, revealing whether models exhibit systematic bias at specific degradation levels.

7.5 Discussion

7.5.1 Effectiveness of Hierarchical Feature Extraction

The finding that the CNN1D model achieved higher accuracy than the recurrent models highlights the effectiveness of hierarchical feature extraction over temporal sequence modelling for this specific 8.5-hour sequence structure. The 6-feature formulation (Voltage, Current, Temperature, OCV, dV/dt, and dI/dt) provides rich, localised physical information. The CNN1D naturally constructs a hierarchical representation of these multi-channel inputs, allowing it to detect complex interactions between the physical derivatives and raw sensor values better than the step-by-step processing of the GRU and LSTM.

Generalization through convolution: Despite traditional expectations that LSTMs and GRUs are better suited for time-series tasks, the rigid profile-level split revealed that recurrent models struggled slightly more with zero-shot generalization to unseen aging trajectories compared to the robust, translation-invariant features learned by the CNN1D. The convolutional filters detect local degradation signatures (e.g., voltage plateau shapes, impedance-related transients) regardless of their temporal position within the 8.5-hour window, making the learned representations inherently more transferable across different aging conditions.

7.5.2 Impact of the Strict Splitting Strategy

The profile-level split strategy is dramatically more challenging than random or cell-level splitting. Under random splitting, samples from the same cell (and thus the same aging trajectory) would appear in both training and test sets, allowing models to achieve artificially high scores by interpolating along known trajectories. The evaluation protocol used here answers the question 'Can the model predict SoH for an aging condition it has never encountered?' rather than 'Can the model interpolate within known aging conditions?'. The former is far more relevant for real-world deployment.

7.5.3 Limitations of the SoH Study

- Single cell chemistry: All results are specific to the NMC/C+SiO LG INR18650HG2 cell. Generalisation to other chemistries would require retraining or transfer learning.
- Fixed window size: The 8.5-hour window may not capture the optimal amount of degradation history.
- Generalisation gap: The approximately 10-point R^2 gap between train and test for the CNN1D suggests a well-balanced capacity, though minor improvements could be explored.
- No frequency-domain features: The models operate entirely in the time domain. Incorporating frequency-domain features could capture periodic degradation patterns.
- No attention mechanisms: Attention-based architectures were not evaluated, though they represent a promising direction for future exploration.

Chapter 8 – Comparative Discussion and Integrated Interpretation

8.1 SoC versus SoH: Task Characteristics

The SoC and SoH estimation tasks differ fundamentally in their temporal scales, input representations, and evaluation protocols, which in turn influence the optimal choice of model architecture.

Table 13: Comparison of the SoC and SoH estimation tasks.

Aspect	SoC Estimation	SoH Estimation
Prediction target	SoC (0–100%)	SoH (50–100%)
Input features	3 (V, I, T)	6 (V, I, T, OCV, dV/dt, dI/dt)
Window length	64 steps (~21 min at 20 s)	256 steps (~8.5 h at 120 s)
Forecast horizon	8 steps (~160 s)	No horizon (current state)
Data split	Cell-level (70/15/15%)	Profile-level (70/15/15%)
SoH filter	SoH ≥ 95%	50% ≤ SoH ≤ 100%
Best model	GRU (RMSE = 0.9513 p.p.)	CNN1D (MAE = 3.31%)
Best R^2	0.9993 (GRU)	0.812 (CNN1D)

8.2 Why Different Architectures Excel

For SoC estimation, the task essentially asks: 'Given the recent electrical trajectory of the cell, what is the SoC 160 seconds from now?' This is a short-horizon, high-frequency regression problem where the target is a smooth, continuously evolving quantity. The input window spans only 21 minutes, and the critical information is the temporal trajectory of voltage, current, and temperature. Recurrent architectures (GRU, LSTM) are naturally suited to this task because they maintain a hidden state that acts as a running summary of the charge-discharge history, effectively performing a learned version of Coulomb counting.

For SoH estimation, the task is fundamentally different: 'Given 8.5 hours of operational data at an unknown point in the cell's life, what is the cell's overall health?' This requires extracting subtle degradation signatures—changes in voltage plateaus, impedance-related voltage drops, temperature response patterns—that may appear anywhere in the window. The CNN1D's translation-invariant filters are well suited to detecting such local patterns regardless of their temporal position. Moreover, the hierarchical architecture allows the CNN to combine low-level features (individual voltage shapes) into higher-level degradation indicators more effectively than the sequential processing of recurrent models.

8.3 The Role of Input Features

The SoC study uses 3 raw features (V, I, T), while the SoH study uses 6 features including OCV and physical derivatives (dV/dt, dI/dt). The inclusion of derivatives in the SoH study is motivated by the fact that degradation manifests through changes in the dynamic response of the cell: an aged cell exhibits larger voltage transients under load (higher internal resistance) and different rate-of-change patterns compared to a fresh cell. These derivative features provide the CNN with explicit rate-of-change information that would otherwise need to be inferred from raw signals.

The SoC study deliberately excludes OCV and prior SoC to ensure that the models provide self-contained estimates from instantaneous measurements only. This design choice is practical: it means the SoC model can serve as an independent correction mechanism for traditional Coulomb-counting-based SoC tracking.

8.4 Evaluation Rigour

The two studies employ different levels of evaluation stringency. The SoC study uses cell-level splitting, ensuring no data from the same cell appears in both training and test sets. The SoH study uses profile-level splitting, ensuring no data from the same aging condition appears in both sets—a strictly harder requirement. This difference partially explains the gap in absolute R^2 values: the SoC models achieve $R^2 > 0.998$, while the SoH models achieve $R^2 \approx 0.81$ at best. The SoH task is inherently harder due to the stricter split and the broader degradation range covered.

8.5 Practical Implications for Battery Management Systems

From a BMS deployment perspective, the combined findings suggest the following architecture recommendations:

- For real-time SoC tracking: Deploy a GRU model with a short input window (~21 minutes). The GRU offers the best accuracy with a simpler architecture than the LSTM, making it suitable for embedded systems.
- For periodic SoH assessment: Deploy a CNN1D model that processes longer windows (~8.5 hours) during scheduled maintenance or check-up periods. The CNN1D's robustness to unseen aging conditions makes it suitable for fleet-level health monitoring.
- Combined deployment: The two models can operate in parallel—the GRU providing continuous SoC estimates and the CNN1D performing periodic health assessment—forming a complementary dual-model BMS architecture.

8.6 Common Findings Across Both Studies

Several findings are consistent across both the SoC and SoH studies:

- All deep learning architectures outperform non-temporal baselines, confirming the importance of temporal modelling.
- Proper data splitting (cell-level or profile-level) is essential for honest evaluation of generalisation.
- The dataset published by Luh and Blank (2024) provides sufficient volume and diversity for meaningful deep learning experiments.
- Regularisation techniques (dropout, weight decay, gradient clipping, early stopping) are critical for preventing overfitting.
- The NMC/C+SiO chemistry introduces complex degradation patterns that can be captured by appropriately designed neural networks.

Chapter 9 – Conclusion and Future Work

9.1 Summary of Contributions

This thesis presented a systematic investigation of deep learning architectures for battery State of Charge and State of Health estimation using the comprehensive Luh and Blank (2024) aging dataset comprising 228 LG INR18650HG2 NMC/C+SiO cells aged under 76 distinct operational conditions.

For SoC estimation, four architectures were compared—MLP, CNN, LSTM, and GRU—using 3 input features (voltage, current, temperature) over 64-timestep windows with a forecast horizon of 8 steps. The GRU achieved the best overall test-set performance with an RMSE of 0.9513 percentage points and an MAE of 0.1614 percentage points. The LSTM followed closely (RMSE = 1.0040, MAE = 0.1847). All models exceeded R^2 = 0.998, but the recurrent architectures demonstrated clear superiority in both quantitative metrics and qualitative prediction quality. The GRU is recommended as the preferred architecture for SoC estimation.

For SoH estimation, three architectures were evaluated—LSTM, GRU, and CNN1D—using 6 input features (V, I, T, OCV, dV/dt, dI/dt) over 256-timestep windows under a strict profile-level evaluation protocol. The CNN1D emerged as the most accurate model with an MAE of 3.31%, an RMSE of 5.65%, and an R^2 of 0.812 on completely unseen test profiles. The GRU achieved an MAE of 3.85% (R^2 = 0.770) and the LSTM an MAE of 4.31% (R^2 = 0.728). The strict profile-level data split ensures that these results reflect genuine generalisation to unseen battery usage conditions.

9.2 Key Findings

- Gated recurrent networks (GRU, LSTM) are well matched to the temporal, integrative nature of SoC dynamics. The GRU offers the best accuracy with a simpler gating mechanism than the LSTM.
- For SoH estimation under strict cross-condition evaluation, the CNN1D's hierarchical temporal filters proved most robust, outperforming the recurrent models on all metrics.

- Different architectures excel at different tasks: recurrent models for short-horizon SoC tracking, convolutional models for longer-horizon SoH assessment.
- The inclusion of physical derivatives (dV/dt, dI/dt) as explicit input features significantly enriches the information available to the SoH models.
- Evaluation rigour matters: the SoH study's profile-level split is dramatically more challenging than cell-level or random splitting, and results should not be directly compared to studies using less rigorous protocols.

9.3 Future Work

Several promising directions emerge from this thesis:

- Attention-augmented architectures: Adding temporal attention mechanisms to the LSTM/GRU (or using full Transformer encoders) could help the models selectively focus on the most informative timesteps.
- Multi-task learning: Jointly predicting SoC, SoH, internal resistance (R_0, R_1), and remaining useful life (RUL) could induce shared representations that improve all tasks.
- Transfer learning across chemistries: Pre-training on the large LG INR18650HG2 dataset and fine-tuning on smaller datasets from different cell chemistries could enable rapid adaptation.
- Extended SoC study to degraded cells: Removing the SoH ≥ 95% filter and evaluating SoC estimation performance across the full degradation range.
- Uncertainty quantification: Implementing Bayesian layers or Monte Carlo Dropout could provide prediction confidence intervals alongside point estimates.
- Data augmentation: Techniques such as time warping, magnitude scaling, and jittering could increase effective training set size and reduce overfitting.
- Explainability: Applying gradient-based attribution methods (e.g., Grad-CAM adapted for 1D signals) could reveal which temporal regions and features contribute most to predictions.

9.4 Lessons Learned

Several practical lessons emerged from this work. First, the importance of data quality and preprocessing cannot be overstated: the timestamp synchronisation challenge in the SoH study required careful engineering to resolve. Second, the choice of data splitting strategy has a dramatic impact on reported metrics and must be clearly documented for fair comparison across studies. Third, simpler models are not always worse: the GRU outperformed the LSTM for SoC (despite fewer parameters), and the CNN1D outperformed both recurrent models for SoH. The right architecture depends on the task structure, not on model complexity alone.

References

[1] Luh, M. and Blank, T. (2024). Comprehensive battery aging dataset: capacity and impedance fade measurements of a lithium-ion NMC/C-SiO cell. Scientific Data, 11, 1004.

[2] Li W, Yang Y, Pischinger S. Domain Generalization Using Maximum Mean Discrepancy Loss for Remaining Useful Life Prediction of Lithium-Ion Batteries. *Batteries*. 2025; 11(5):194

[3] Plett, G. L. (2004). Extended Kalman filtering for battery management systems of LiPB-based HEV battery packs: Part 1–3. Journal of Power Sources, 134(2), 252–292.

[4] Cho, K., van Merrienboer, B., Gulcehre, C., et al. (2014). Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation. Proceedings of EMNLP, 1724–1734.

[5] LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep Learning. Nature, 521, 436–444.

[6] Zhang, Y., Xiong, R., He, H., and 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.

[7] Chemali, E., Kollmeyer, P. J., Preindl, M., and Emadi, A. (2018). State-of-Charge Estimation of Li-ion Batteries Using Deep Neural Networks. Journal of Power Sources, 400, 242–255.

[8] Shen, S., Sadoughi, M., Li, M., Wang, Z., and Hu, C. (2020). Deep convolutional neural networks with ensemble learning and transfer learning for capacity estimation of lithium-ion batteries. Applied Energy, 260, 114296.

[9] Richardson, R. R., Birkl, C. R., Osborne, M. A., and Howey, D. A. (2019). Gaussian Process Regression for In Situ Capacity Estimation of Lithium-Ion Batteries. IEEE Transactions on Industrial Informatics, 15(1), 127–138.

[10] Kingma, D. P. and Ba, J. (2015). Adam: A Method for Stochastic Optimisation. Proceedings of ICLR.

[11] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: A Simple Way to Prevent Neural Networks from Overfitting. JMLR, 15, 1929–1958.

[12] Severson, K. A., Attia, P. M., Jin, N., et al. (2019). Data-driven prediction of battery cycle life before capacity degradation. Nature Energy, 4, 383–391.

[13] Birkl, C. R., Roberts, M. R., McTurk, E., Bruce, P. G., and Howey, D. A. (2017). Degradation diagnostics for lithium ion cells. Journal of Power Sources, 341, 373–386.

[14] Paszke, A., Gross, S., Massa, F., et al. (2019). PyTorch: An Imperative Style, High-Performance Deep Learning Library. NeurIPS, 32.

[15] Edge, J. S. et al. (2021). Lithium ion battery degradation: what you need to know. Physical Chemistry Chemical Physics, 23, 8200–8221.

[16] Vermeer, W., Chandra Mouli, G. R., and Bauer, P. (2022). A Comprehensive Review on the Characteristics and Modeling of Lithium-Ion Battery Aging. IEEE TTE, 8, 2205–2232.

[17] Yang, X.-G., Leng, Y., Zhang, G., Ge, S., and Wang, C.-Y. (2017). Modeling of lithium plating induced aging of lithium-ion batteries. Journal of Power Sources, 360, 28–40.

[18] Fichtner, M. (2022). Recent Research and Progress in Batteries for Electric Vehicles. Batteries & Supercaps, 5, e202100224.

[19] LG Chem (2014). Technical Information of LG 18650HG2 (3.0 Ah).

[20] Brodsky Ringler, P. et al. (2023). Modeling of Lithium Plating and Stripping Dynamics during Fast Charging. Batteries, 9, 337.

[21] El Mejdoubi, Asmae, Amrane Oukaour, Hicham Chaoui, Hamid Gualous, Jalal Sabor, and Youssef Slamani. "State-of-charge and state-of-health lithium-ion batteries' diagnosis according to surface temperature variation." *IEEE Transactions on Industrial Electronics* 63, no. 4 (2015): 2391-2402.

[22] Xiong, Rui, Linlin Li, and Jinpeng Tian. "Towards a smarter battery management system: A critical review on battery state of health monitoring methods." *Journal of Power Sources* 405 (2018): 18-29.

[23] Song, Hyunsik, and Donghyun Shin. "Method for evaluating the accuracy of state-of-charge (SOC)/state-of-health (SOH) estimation of BMSs." *Energy Science & Engineering* 11, no. 11 (2023): 4273-4286.

[24] Song, Yuchen, Datong Liu, Haitao Liao, and Yu Peng. "A hybrid statistical data-driven method for on-line joint state estimation of lithium-ion batteries." *Applied Energy* 261 (2020): 114408.

[25] Topan, Paris Ali, M. Nisvo Ramadan, Ghufron Fathoni, Adha Imam Cahyadi, and Oyas Wahyunggoro. "State of Charge (SOC) and State of Health (SOH) estimation on lithium polymer battery via Kalman filter." In *2016 2nd International Conference on Science and Technology-Computer (ICST)*, pp. 93-96. IEEE, 2016.

[26] Tran, Manh-Kien, Manoj Mathew, Stefan Janhunen, Satyam Panchal, Kaamran Raahemifar, Roydon Fraser, and Michael Fowler. "A

comprehensive equivalent circuit model for lithium-ion batteries, incorporating the effects of state of health, state of charge, and temperature on model parameters." *Journal of Energy Storage* 43 (2021): 103252.

[27] Kassim, Mohamed Rawidean Mohd, Wan Adil Wan Jamil, and Roslee Mohd Sabri. "State-of-charge (soc) and state-of-health (soh) estimation methods in battery management systems for electric vehicles." In *2021 IEEE International Conference on Computing (ICOCO)*, pp. 91-96. IEEE, 2021.

[28] Murnane, Martin, and Adel Ghazel. "A closer look at state of charge (SOC) and state of health (SOH) estimation techniques for batteries." *Analog devices* 2 (2017): 426-436.

[29] Gao, Yizhao, Kailong Liu, Chong Zhu, Xi Zhang, and Dong Zhang. "Co-estimation of state-of-charge and state-of-health for lithium-ion batteries using an enhanced electrochemical model." *IEEE Transactions on Industrial Electronics* 69, no. 3 (2021): 2684-2696.

[30] Moura, Scott J., Nalin A. Chaturvedi, and Miroslav Krstić. "Adaptive partial differential equation observer for battery state-of-charge/state-of-health estimation via an electrochemical model." *Journal of Dynamic Systems, Measurement, and Control* 136, no. 1 (2014): 011015.

[31] Wassiliadis, Nikolaos, Jörn Adermann, Alexander Frericks, Mikhail Pak, Christoph Reiter, Boris Lohmann, and Markus Lienkamp. "Revisiting the dual extended Kalman filter for battery state-of-charge and state-of-health estimation: A use-case life cycle analysis." *Journal of Energy Storage* 19 (2018): 73-87.

[32] Xu, Zhicheng, Jun Wang, Peter D. Lund, and Yaoming Zhang. "Co-estimating the state of charge and health of lithium batteries through combining a minimalist electrochemical model and an equivalent circuit model." *Energy* 240 (2022): 122815.

[33] Amir, Shehla, Moneeba Gulzar, Muhammad O. Tarar, Ijaz H. Naqvi, Nauman A. Zaffar, and Michael G. Pecht. "Dynamic equivalent circuit model to estimate state-of-health of lithium-ion batteries." *Ieee Access* 10 (2022): 18279-18288.

[34] Shen, Ping, Minggao Ouyang, Languang Lu, Jianqiu Li, and Xuning Feng. "The co-estimation of state of charge, state of health, and state of function for lithium-ion batteries in electric vehicles." *IEEE Transactions on vehicular technology* 67, no. 1 (2017): 92-103.

[35] Tong, Shijie, Matthew P. Klein, and Jae Wan Park. "On-line optimization of battery open circuit voltage for improved state-of-charge and state-of-health estimation." *Journal of Power Sources* 293 (2015): 416-428.

[36] Tan, Xiaojun, Yuqing Tan, Di Zhan, Ze Yu, Yuqian Fan, Jianzhi Qiu, and Jun Li. "Real-time state-of-health estimation of lithium-ion batteries based on the equivalent internal resistance." *IEEE Access* 8 (2020): 56811-56822.

[37] Vidal, Carlos, Pawel Malysz, Phillip Kollmeyer, and Ali Emadi. "Machine learning applied to electrified vehicle battery state of charge and state of health estimation: State-of-the-art." *IEEE Access* 8 (2020): 52796-52814.

[38] Sepasi, Saeed, Reza Ghorbani, and Bor Yann Liaw. "Improved extended Kalman filter for state of charge estimation of battery pack." *Journal of Power Sources* 255 (2014): 368-376.

[39] Ahmed, Ryan, Mohammed El Sayed, Ienkaran Arasaratnam, Jimi Tjong, and Saeid Habibi. "Reduced-order electrochemical model parameters identification and state of charge estimation for healthy and aged Li-ion batteries—Part II: Aged battery model and state of charge estimation." *IEEE Journal of Emerging and Selected Topics in Power Electronics* 2, no. 3 (2014): 678-690.

[40] Soyoye, Babatunde D., Indranil Bhattacharya, Mary Vinolisha Anthony Dhason, and Trapa Banik. "State of charge and state of health estimation in electric vehicles: Challenges, approaches and future directions." *Batteries* 11, no. 1 (2025): 32.

[41] Remmlinger, Jürgen, Michael Buchholz, Markus Meiler, Peter Bernreuter, and Klaus Dietmayer. "State-of-health monitoring of lithium-ion batteries in electric vehicles by on-board internal resistance estimation." *Journal of power sources* 196, no. 12 (2011): 5357-5363.

[42] Ren, Zhong, and Changqing Du. "A review of machine learning state-of-charge and state-of-health estimation algorithms for lithium-ion batteries." *Energy Reports* 9 (2023): 2993-3021.

[43] Ma, Lili, Yonghong Xu, Hongguang Zhang, Fubin Yang, Xu Wang, and Cheng Li. "Co-estimation of state of charge and state of health for lithium-ion batteries based on fractional-order model with multi-innovations unscented Kalman filter method." *Journal of Energy Storage* 52 (2022): 104904.

[44] Shibl, Mostafa M., Loay S. Ismail, and Ahmed M. Massoud. "A machine learning-based battery management system for state-of-charge prediction and state-of-health estimation for unmanned aerial vehicles." *Journal of Energy Storage* 66 (2023): 107380.

[45] Dannier, Adolfo, Gianluca Brando, Mattia Ribera, and Ivan Spina. "Li-ion batteries for electric vehicle applications: An overview of accurate state of charge/state of health estimation methods." *Energies* 18, no. 4 (2025): 786.

[46] Huang, Zexin, Matt Best, James Knowles, and Ashley Fly. "Adaptive piecewise equivalent circuit model with SOC/SOH estimation based on extended Kalman filter." *IEEE Transactions on Energy Conversion* 38, no. 2 (2022): 959-970.

[47] Dey, Satadru, Beshah Ayalew, and Pierluigi Pisu. "Combined estimation of state-of-charge and state-of-health of Li-ion battery cells using SMO on electrochemical model." In *2014 13th International Workshop on Variable Structure Systems (VSS)*, pp. 1-6. IEEE, 2014.

[48] Zou, Yuan, Xiaosong Hu, Hongmin Ma, and Shengbo Eben Li. "Combined state of charge and state of health estimation over lithium-ion battery cell cycle lifespan for electric vehicles." *Journal of Power Sources* 273 (2015): 793-803.

[49] Lai, Xin, Shuyu Wang, Shangde Ma, Jingying Xie, and Yuejiu Zheng. "Parameter sensitivity analysis and simplification of equivalent circuit model for the state of charge of lithium-ion batteries." *Electrochimica Acta* 330 (2020): 135239.

[50] Kumar, R. Ranjith, Chokkalingam Bharatiraja, K. Udhayakumar, S. Devakirubakaran, K. Sathiya Sekar, and Lucian Mihet-Popa. "Advances in batteries, battery modeling, battery management system, battery thermal management, SOC, SOH, and charge/discharge characteristics in EV applications." *Ieee Access* 11 (2023): 105761-105809.

Appendix A – Additional SoH Figures

This appendix collects supplementary visual results for the SoH estimation study.

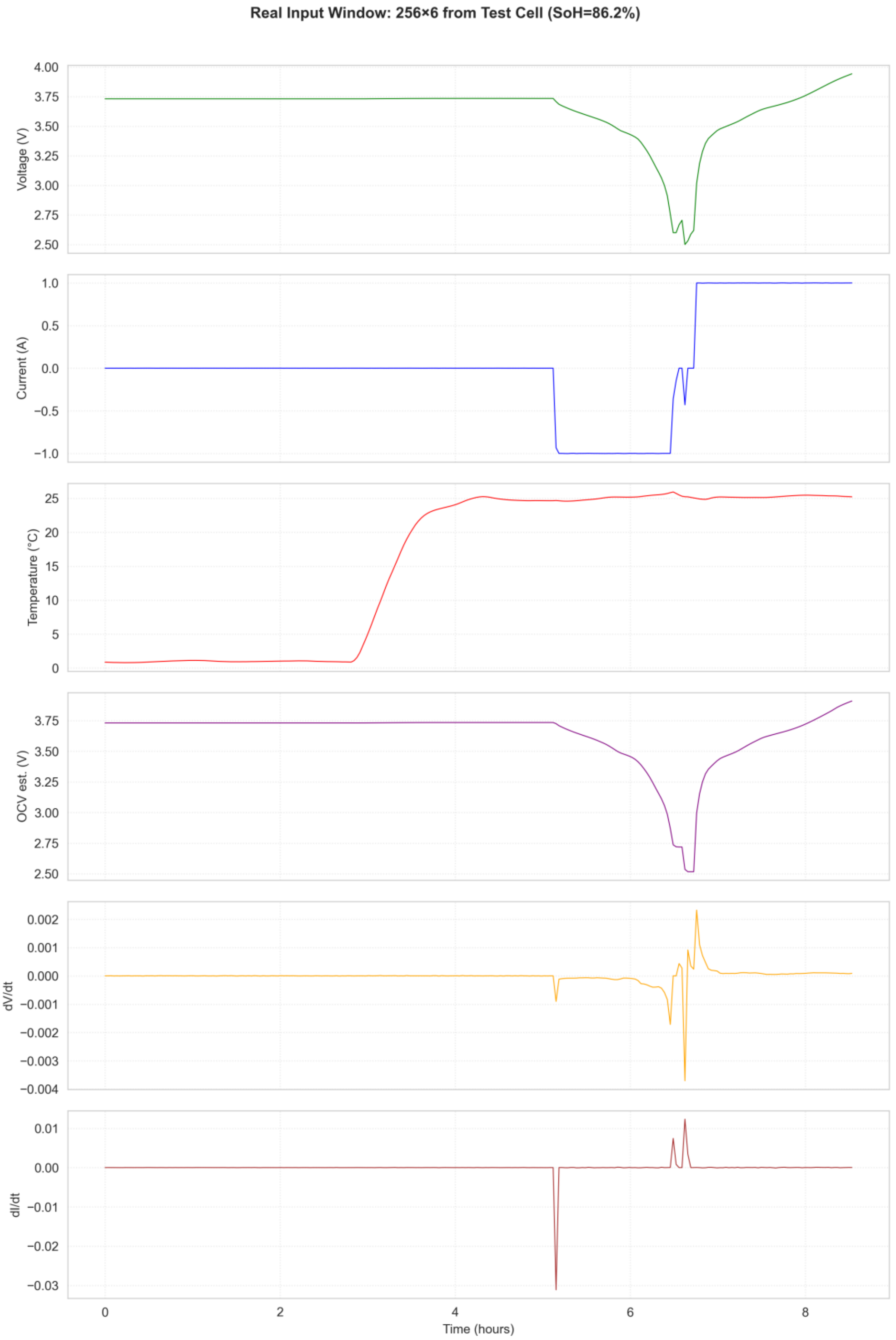

Figure 14: Alternative 4-channel input window view for SoH estimation.

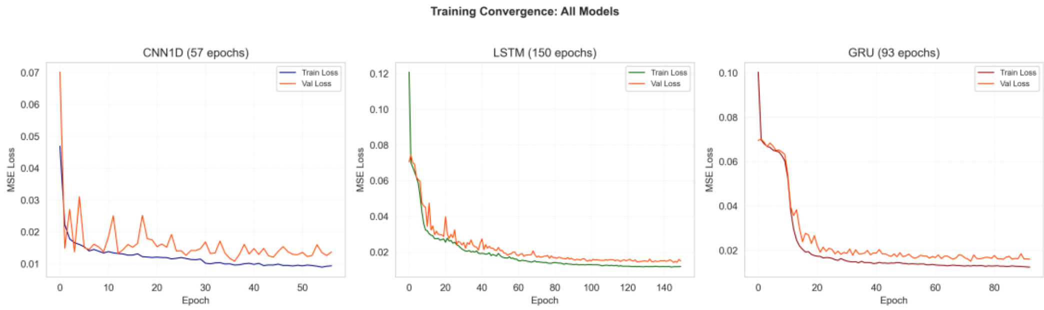

Figure 15: Extended training curves for all SoH models.

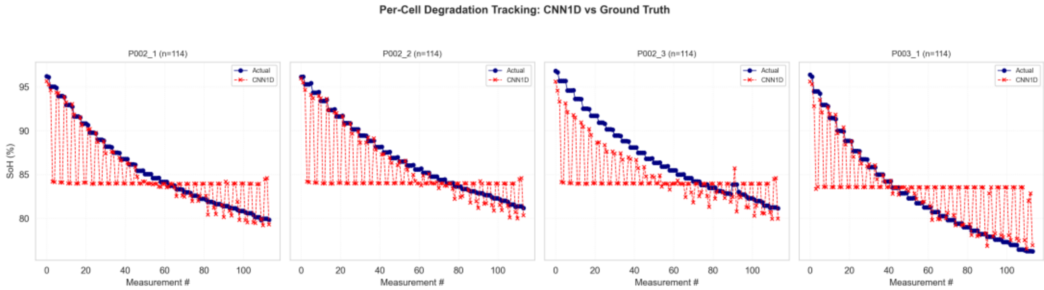

Figure 16: Per-cell SoH tracking across the aging experiment.

Figure 17: Detailed regression and residual analysis for SoH models.

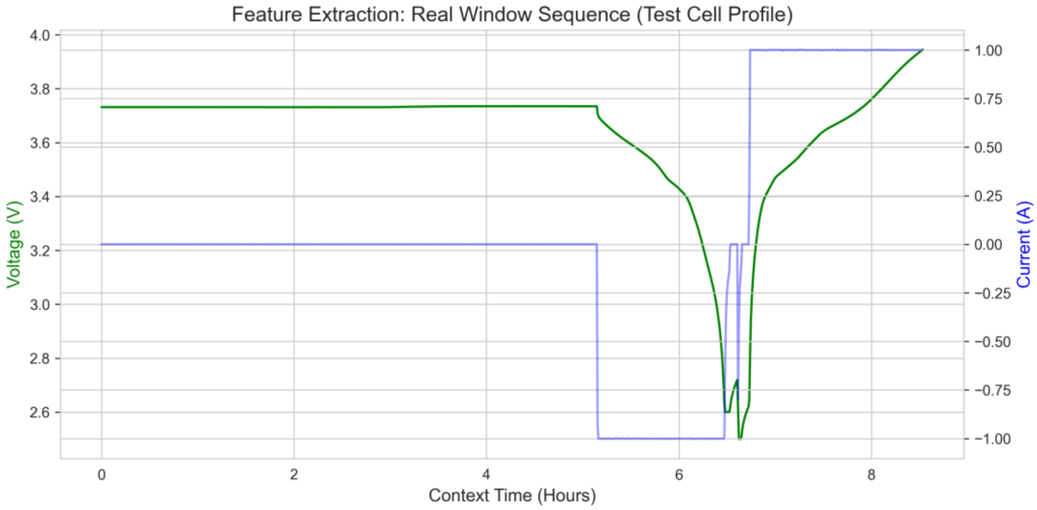

Figure 18: Sequential sample windows extracted for SoH estimation.