Predicting Thermal Runaway of Li-ion Battery in EV using Deep Learning (DL) and Internet of Things (IoT)

Submitted in partial fulfillment of the requirements

of

The Degree of

Bachelor of Technology

By

Saniya Jena

(Roll No. 231220051)

Under the supervision of:

Dr. (Prof.) Jyoteesh Malhotra

pursued in

Department of Electronics and Communication

NATIONAL INSTITUTE OF TECHNOLOGY DELHI

(Session 2023-2027)

Approval Sheet

This Project work entitled "**Predicting Thermal Runaway of Li-ion Battery in EV using Deep Learning (DL) and Internet of Things (IoT)**" by **Ms. Saniya Jena** is approved for the degree of **Bachelor of Technology** in **Electronics and Communication**.

Examiners:

Dr. Rikmantra Basu, Dr. Manish Verma, Dr. Preeti Verma

Supervisor:

Dr. (Prof.) Jyoteesh Malhotra

Date: _____

Place: _____

Certificate

This is to certify that the dissertation entitled "**Predicting Thermal Runaway of Li-ion Battery in EV using Deep Learning (DL) and Internet of Things (IoT)**" submitted to the **Department of Electronics and Communication, National Institute of Technology**, GT Karnal Road, Delhi-110036, in partial fulfillment of the requirements for the degree of **B.Tech. (Electronics & Communication) (Session 2023-2027)** is a record of original work done by **Ms. Saniya Jena (Roll No. 231220051)** during the academic year 2024-2025 under the Supervision of **Dr. (Prof.) Jyoteesh Malhotra**, Department of Electronics and Communication, National Institute of Technology, Delhi. The dissertation has not formed the basis for the award of any Degree.

(Signature of the Supervisor)

DR. (PROF.) JYOTEESH MALHOTRA

Department of Electronics and Communication

National Institute of Technology, Delhi

Declaration

I declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources, which have thus not been properly cited, or from whom proper permission has not been taken when needed.

(Signature of Student)

Saniya Jena

Roll No: 231220051

Date: _____

Acknowledgements

We would like to express our sincere gratitude to **Professor Ajay K. Sharma**, Director, National Institute of Technology Delhi, for providing a stimulating academic environment and the necessary resources that greatly facilitated the successful completion of this project. We would like to express our sincere gratitude to **Prof. (Dr.) Jyoteesh Malhotra**, Professor, Department of Electronics and Communication Engineering, **National Institute of Technology Delhi**, for his invaluable guidance, support, and encouragement throughout the course of this project. His expertise and constructive feedback played a crucial role in shaping the direction and quality of this work. We extend our heartfelt thanks to the **Head of Department**, faculty members, and staff of the Department of Electronics and Communication Engineering for their continuous support and for providing the necessary facilities during the project. Lastly, We are deeply grateful to our families for their unwavering support, patience, and motivation throughout this academic journey.

Abstract

Thermal runaway in lithium-ion battery packs is one of the most hazardous failure modes in electric vehicles (EVs), where unchecked internal heat generation can lead to fires or explosions. To address this, we propose a prognostic framework integrating signal processing, deep learning, and IoT connectivity to detect early signs of abnormal heat generation (AHG) well before traditional safety systems respond.

Our system collects high-frequency measurements via an Arduino-based sensor network, including cell voltage, current, temperatures from six thermocouples, mechanical displacement, and ambient data. Time series data are cleaned using rolling window statistics, Savitsky–Golay filtering, and lagged feature generation. PCA reduces dimensionality while preserving over 97% variance, minimizing computational load.

At the core is a CNN–LSTM model. Convolutional layers extract spatial patterns across sensor inputs, while peephole LSTM cells with attention capture long-term temporal dependencies and emphasize key sequence segments. Using a 48-step sliding window, the model predicts future temperature trajectories, enabling real-time heat generation proxy calculations. These are compared with analytical Qnormal estimates, triggering alerts when AHG exceeds thresholds.

A Random Adjacent Optimization Method (RAOM) dynamically adjusts hyperparameters—window size, filter count, and kernel width—based on validation feedback, removing the need for manual grid searches.

Tested on lab datasets, our framework achieves an average MAE of 1.9783 °C and RMSE of 2.3223 °C across thermocouples, with over 35 seconds of lead time before runaway conditions. Real-time data are streamed to a cloud-hosted inference engine that issues AHG alerts to dashboards or fleet systems.

By combining rigorous data preprocessing, deep learning automation, and real-time IoT integration, this work offers a scalable and reliable solution for early thermal runaway detection in EV batteries—enhancing operational safety and building driver confidence.

TABLE OF CONTENTS

Chapter 1: Introduction

The rapid adoption of electric vehicles (EVs) hinges on their performance and range and the reliability and safety of their lithium-ion battery systems. Among these batteries' most severe failure modes is thermal runaway, a self-accelerating event in which internal heat generation outpaces dissipation, leading to runaway temperature increases, cell rupture, and potentially catastrophic fires. Early detection of abnormal heat generation (AHG) within battery cells is paramount to preventing thermal runaway and ensuring passenger safety.

In this project, we develop an end-to-end deep learning–IoT framework for multi-step prognosis of thermal runaway in EV battery packs. First, we preprocess voltage, current, temperature, and mechanical displacement signals with rolling statistics and Savitsky–Golay smoothing to reduce noise and enhance thermal feature stability. We then compress these inputs via principal component analysis (PCA) and construct sliding-window sequences to capture both short-term dynamics and longer-term trends in battery behaviour.

Our core prediction engine is a hybrid convolutional neural network (CNN) and long short-term memory (LSTM) architecture, augmented with a custom peephole LSTM cell and attention mechanism. This CNN–LSTM model forecasts future temperature trajectories across six cell locations over a 48-step horizon. It allows us to compute real-time heat generation proxies and compare them against normal heat dissipation estimates. Rather than manually tuning hyperparameters, we employ a random adjacent optimization method (RAOM) to automatically discover optimal window sizes, filter depths, and kernel widths—improving forecast accuracy and eliminating labour-intensive grid searches.

To bring this prognostic capability into operational settings, we integrate IoT-enabled battery management modules that stream live sensor data to a cloud-hosted inference service. When predicted temperatures exceed safety thresholds or abnormal heat generation surpasses calibrated limits, instantaneous alerts are issued to drivers or fleet operators, granting vital lead time—measured in minutes—to initiate cooling interventions or controlled shutdowns. By combining advanced signal processing, automated deep learning, and real-time IoT connectivity, this project delivers a robust, scalable solution for battery thermal runaway prognosis in electric vehicles, paving the way for safer, more reliable EV deployments.

Chapter 2: Literature Review

Battery thermal runaway prediction and diagnosis have attracted significant research interest due to the safety-critical nature of lithium-ion systems in electric vehicles (EVs) and energy storage applications. Early data-driven approaches leveraged clustering and classical machine learning to identify abnormal thermal behaviour. DBSCAN was utilized by Sun *et al.* to detect potential thermal runaway cells based on voltage deviation and cumulative deviation features, demonstrating high accuracy in pinpointing anomalies in real-world EV datasets [1]. Similarly, ensemble learning methods—random forest, GBDT, and XGBoost—were applied by Li *et al.* to predict thermal runaway risks, achieving a recognition rate of 66.7% and 5–7 minutes of advance warning, though with a 16.5% false-alarm rate that highlights the need for improved robustness [2].

More recent studies have adopted hybrid and deep learning frameworks to enhance both accuracy and prognostic horizon. A modified CNN-LSTM architecture was proposed in [3] to forecast battery temperature up to eight minutes ahead, achieving a mean relative error (MRE) of 0.28% and extending prognosis to 27 minutes using abnormal heat generation rates and PCA for feature compression. Adversarial Invariance Encoding (AIDE) was introduced by Chen *et al.* to address data paucity and domain generalization, improving prediction performance by 128.6% over state-of-the-art models across unseen battery materials and states-of-charge [4]. An improved LSTM–Hidden Markov Model (LSTM-HMM) combining bidirectional LSTM and GMM-HMM demonstrated 98.68% accuracy and real-time response in under 0.1 ms, though it remains to integrate external factors such as ambient conditions and aging [5].

Hybridizing data-driven techniques with physics-informed or clustering methods has also shown promise. In [6], XGBoost-PCA-DBSCAN integrated dimensionality reduction and density-based clustering to provide 35 min advance warning with seasonal mean square errors between 0.0523 and 0.0747, outperforming standard XGBoost by over 31%. However, adaptation to diverse battery chemistries and vehicle environments remains an open challenge. Tian *et al.* developed an SVM-based diagnosis framework optimized via Bayesian methods, achieving ~90% accuracy in classifying thermal runaway modes and identifying heat loss rate as the most critical feature, yet lacking real-time IoT integration [7]. More holistic system-level approaches have begun to emerge: Wang *et al.* incorporated thermal energy storage (TES) with machine learning (logistic regression, SVM, Naïve Bayes) to both predict and mitigate

runaway risks under vehicle-to-grid scenarios, reaching 98% accuracy but requiring validation across broader TES configurations and experimental conditions [8].

Despite these advances, key gaps persist: most models focus on single-cell behaviour without fully accounting for cell-to-cell coupling or pack-level dynamics; data scarcity and domain shift across chemistries and operating profiles continue to hinder generalization; and integration of physics-based thermal models with fast, on-board prognostic algorithms remains an open research avenue. Future work must therefore pursue hybrid physics-data approaches, enrich datasets via adversarial or transfer learning, and validate prognostic methods in real-world EV systems.

Chapter 3: Components Used

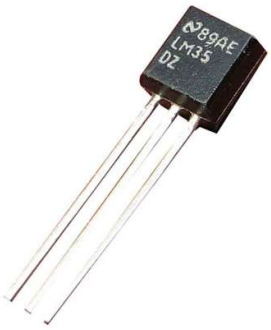
1. **VL53L0X displacement sensor**

The **VL53L0X displacement sensor** plays a critical role in detecting physical deformation of the battery casing, which is often caused by gas buildup during the early stages of thermal runaway. This sensor operates using Time-of-Flight (ToF) technology, where it emits an infrared laser and calculates the time taken for the reflected signal to return. Based on this, it accurately determines the distance between the sensor and the battery surface. With a measurement range of 30 mm to 2000 mm, it can effectively detect even slight expansions of the battery casing, helping predict possible failure before it escalates.

2. **Force Sensitive Resistor (FSR402)**

The **Force Sensitive Resistor (FSR)** is used to detect external stress and swelling outside the battery pack. This sensor changes its resistance based on the force applied to it, and this change is translated into a voltage output using a simple voltage divider circuit. As internal pressure builds up due to gas generation or electrode expansion—common in failing batteries—the FSR registers this as a measurable change in pressure. This mechanical feedback is essential for identifying physical battery distress early.

3. **LM35 temperature sensor**



To monitor ambient environmental conditions, the **LM35 temperature sensor** is employed. It provides an analog voltage output that is directly proportional to the surrounding temperature, with a typical scale factor of 10 mV per °C. The LM35 has a wide operating range from -55°C to +150°C and does not require external calibration, making it a reliable choice for monitoring the thermal environment around the battery. Ambient temperature is an important factor in battery safety, as excessive heat from the surroundings can contribute to or accelerate thermal runaway.

4. **DS18B20 digital temperature sensor**

For direct measurement of the battery cell's temperature, the system incorporates the **DS18B20 digital temperature sensor**. This sensor provides accurate real-time thermal data via a 1-Wire digital interface and supports a temperature range from -55°C to +125°C. With a typical accuracy of ±0.5°C over the most relevant range, it is highly effective for monitoring heating trends that may indicate the onset of thermal runaway. Being in direct contact with the battery, the DS18B20 allows for precise thermal profiling during charge and discharge cycles.

5. **B25 voltage sensor**

Lastly, the **B25 voltage sensor** is used to track voltage levels across the battery terminals. This sensor uses a voltage divider circuit to scale down the input voltage (up to 25V) to a microcontroller-safe level. By continuously monitoring voltage behaviour, this sensor can detect anomalies such as overcharging, deep discharge, or short circuits—all of which are electrical precursors to thermal instability. Timely detection of such abnormalities is critical in activating preventive measures before damage becomes irreversible.

6. ESP32

All the sensors are interfaced with an ESP32 microcontroller, which serves as the central processing unit of the monitoring system. The ESP32 is chosen for its high performance, dual-core processor, low power consumption, and built-in Wi-Fi and Bluetooth capabilities, making

it ideal for real-time, wireless IoT-based battery monitoring applications. It reads analog and digital data from the connected sensors, processes it locally, and transmits the information to a cloud server or local dashboard for continuous monitoring and alert generation. The ESP32's multiple GPIO pins, ADC channels, and digital communication support (including I2C, SPI, and UART) ensure seamless integration with sensors like the DS18B20 (1-Wire), LM35 (analog), VL53L0X (I2C), and the B25 voltage divider. Its flexibility and wireless connectivity make it a robust platform for deploying predictive safety models in lithium-ion battery systems.

Chapter 4: Software And Tools

A. Development Environments

- **Google Colab:**

  Google Colab is a cloud-based Jupyter notebook environment that allows you to write and execute Python code in your browser with zero configuration. For your project, you leveraged Colab's built-in free GPU (and optional TPU) acceleration to dramatically reduce training time for your deep learning models. Colab seamlessly mounts your Google Drive, so you were able to store large sensor datasets and intermediate model checkpoints alongside your code. The notebook interface also let you interleave narrative text (with LaTeX support), code cells, and inline plots—ideal for exploratory data analysis, hyperparameter tuning, and documenting your workflow in a single shareable file.

- **Arduino IDE (v1.8.x):**

  The Arduino Integrated Development Environment (IDE) is a cross-platform application used for writing, compiling, and uploading C/C++ code to Arduino microcontrollers. In version 1.8.x, you benefit from a compact editor with syntax highlighting, automatic bracket pairing, and a one-click upload button. You wrote firmware that sampled sensor readings (e.g., temperature, humidity, or light) at fixed intervals, buffered them, and sent them over USB to your host PC. The IDE's built-in Serial Monitor provided a live console for debugging: you could print raw ADC values

or formatted JSON and immediately verify timing, sensor calibration, and data integrity before ingesting the stream into your Python pipeline.

B. Programming Languages

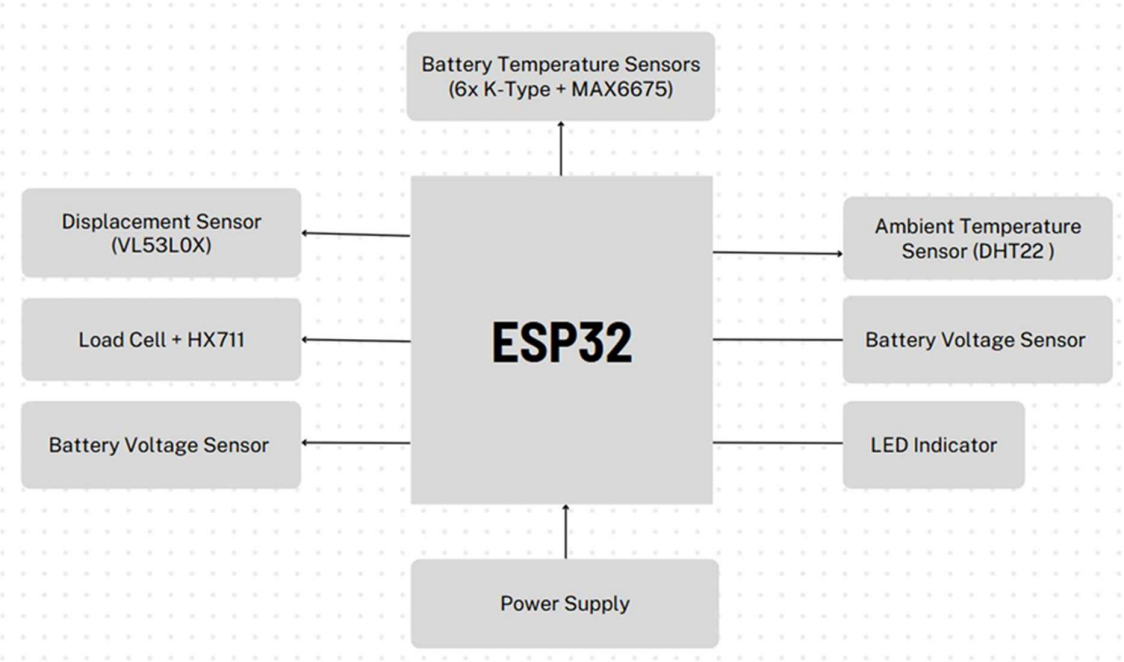
- **Python 3.x**: Served as the primary language for data preprocessing, model implementation, training, and analysis within the Colab environment.
- **C/C++:** Used for developing firmware to interface with sensors and handle real-time data acquisition on the Arduino platform.

C. Python Libraries and Frameworks

- **NumPy and Pandas**: Provided efficient data structures and functions for numerical computations and data manipulation.
- **SciPy:** Applied for signal processing tasks, including Savitsky–Golay smoothing to reduce noise in sensor data.
- **scikit-learn**: Utilized for data preprocessing techniques such as standardization (StandardScaler) and dimensionality reduction (PCA).
- **TensorFlow 2.x (Keras API):** Implemented for constructing and training the deep learning model, incorporating CNN, custom LSTM cells, and attention mechanisms.
- **Matplotlib:** Employed for visualizing training progress, model predictions, and performance metrics through plots and graphs.

Chapter 5: Methodology

This methodology leverages IoT-enabled multi-modal sensing to capture early indicators of thermal runaway, transmits real-time data to a cloud BMS, applies rigorous preprocessing (interpolation, smoothing, feature engineering, PCA) to construct robust inputs, and employs a CNN–LSTM architecture with self-attention and custom LSTM cells. Hyperparameters are tuned via Random Approximation Optimization (RAOM) informed by metaheuristic studies, and the model outputs both multi-step temperature forecasts and binary AHG alerts. Performance is validated with MAE, RMSE, and R^2 metrics, and the final system is containerized for real-time cloud deployment with horizontal scalability.



1. Data Acquisition via IoT Sensors

- Displacement (VL53L0X): Measures battery casing expansion (30 mm–2 m range) to detect gas buildup and early runaway signs.
- Force Sensitive Resistor (FSR402): Captures external pressure variations from swelling or mechanical stress.
- Ambient Temperature (LM35): Records environmental temperature (–55 °C to +150 °C) to contextualize external thermal effects.
- Battery Temperature (DS18B20): Monitors cell surface temperature (–55 °C to +125 °C) for direct overheating detection.
- Voltage (B25): Tracks voltage fluctuations (0 V–25 V range) to identify overcharge/discharge anomalies.

A local ESP32-based BMS unit integrates these sensors, balancing, and security logic, then transmits measurements continuously for centralized analysis.

2. Data Transmission to Cloud-Based BMS

Sensor data are packaged via MQTT over secure TLS channels from the ESP32 to a cloud platform, where a microservices architecture ingests, stores (NoSQL time-series DB), and

preprocesses incoming streams. This layer decouples data ingestion, analytics, and visualization, enabling low-latency alerts and scalable machine learning pipelines.

3. Data Preprocessing & Feature Engineering

1. Cleaning & Imputation: Convert all columns to numeric, apply linear interpolation, then bidirectional forward/backfill to handle remaining NaNs.
2. Denoising: Apply a Savitzky–Golay filter (7-point, polyorder 2) to each temperature channel to smooth high-frequency noise.
3. Temporal Features: Generate lag features (lags 1–3) and 5-point rolling mean/std for all channels to capture short-term dynamics, which have been shown to improve RUL and anomaly prediction in LIBs.
4. Dimensionality Reduction: Standardize features and apply PCA to retain >97% variance, reducing computational complexity and multicollinearity.

4. Model Architecture

- CNN Encoder: Two stacked 1D Conv layers (relu activation, batch norm, dropout 0.3) followed by max-pooling to extract spatial patterns in multi-sensor sequences.
- LSTM with Peepholes: A custom ModifiedLSTMCell integrates cell-state peephole connections to enhance gate control and long-term dependency learning.
- Self-Attention: An Attention layer refines the encoder output into a context vector, improving focus on critical time steps, as in CNN-BiLSTM-AM models MDPI.
- Decoder: A RepeatVector–LSTM stack generates 48-step forecasts for the six temperature channels.
- Safety Logic: A Lambda layer transforms scaled outputs back to real temperatures; ΔT is computed relative to the last observation and summed to form Q_real. Parallel Q_normal is computed via Joule heating (v^2/R_int). Binary alerts fire if max temp > 80 °C or Q_real – Q_normal exceeds threshold, enabling dual-criterion early warning.

Model: "functional"

Layer (type)	Output Shape	Param #	Connected to
input_layer (InputLayer)	(None, 48, 5)	0	-
conv1d (Conv1D)	(None, 48, 32)	512	input_layer[0][0]
batch_normalization (BatchNormalizatio…	(None, 48, 32)	128	conv1d[0][0]
dropout (Dropout)	(None, 48, 32)	0	batch_normalizat…
conv1d_1 (Conv1D)	(None, 48, 32)	3,104	dropout[0][0]
batch_normalizatio… (BatchNormalizatio…	(None, 48, 32)	128	conv1d_1[0][0]
dropout_1 (Dropout)	(None, 48, 32)	0	batch_normalizat…
max_pooling1d (MaxPooling1D)	(None, 24, 32)	0	dropout_1[0][0]
conv1d_2 (Conv1D)	(None, 24, 64)	6,208	max_pooling1d[0]…
batch_normalizatio… (BatchNormalizatio…	(None, 24, 64)	256	conv1d_2[0][0]
dropout_2 (Dropout)	(None, 24, 64)	0	batch_normalizat…
rnn (RNN)	[(None, 64), (None, 64), (None, 64)]	45,312	dropout_2[0][0]
repeat_vector_1 (RepeatVector)	(None, 48, 64)	0	rnn[0][1]
lstm (LSTM)	(None, 48, 64)	33,024	repeat_vector_1[… rnn[0][1], rnn[0][2]
dropout_3 (Dropout)	(None, 48, 64)	0	lstm[0][0]
time_distributed (TimeDistributed)	(None, 48, 6)	390	dropout_3[0][0]

Total params: 89,062 (347.90 KB)
Trainable params: 88,806 (346.90 KB)
Non-trainable params: 256 (1.00 KB)

5. Hyperparameter Optimization

Key hyperparameters (kernel size, filter count, window length) are tuned via a Random Approximation Optimization Method (RAOM), which explores adjacent options in each iteration and adopts configurations that reduce validation loss. This approach parallels metaheuristic optimization techniques (e.g., PSO, Bayesian methods) shown to accelerate convergence in CNN–LSTM models.

6. Prediction of Abnormal Heat Generation & Alerts

Upon each input window, the model returns both the 48-step temperature forecast and a boolean "alert" flag. The alert is raised if either (a) any predicted temperature exceeds 80 °C (thermal threshold) or (b) the predicted abnormal heat generation (Q_real – Q_normal) surpasses the anomaly threshold G = 1000 W, facilitating proactive interventions.

7. Evaluation Metrics & Validation

Model performance is evaluated on a hold-out validation set using:

- MAE & RMSE: To quantify average and squared error magnitudes.

8. Deployment & Real-Time Monitoring

The final CNN–LSTM model is containerized (Docker/Kubernetes) within the cloud BMS, subscribing to the live data stream. Horizontal autoscaling ensures throughput under varying fleet sizes. REST and WebSocket APIs expose forecasts and alerts to driver dashboards and fleet management systems, following CIBMS patterns in recent IoT-cloud integrations and edge-to-cloud LSTM deployments.

Chapter 6: Testing And Results

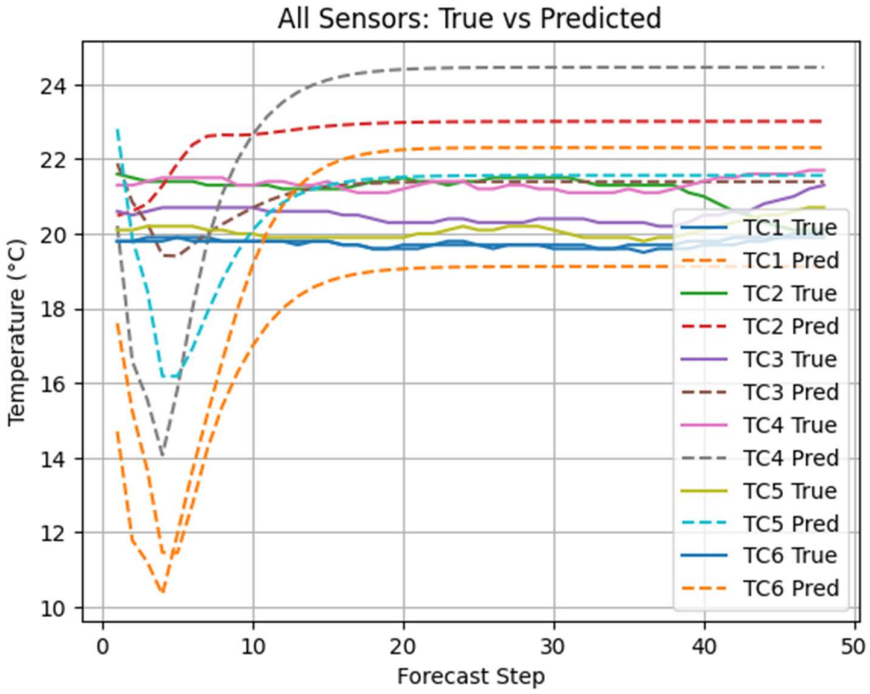
Our CNN–LSTM prognostic model was evaluated on a held-out validation set of real-time sensor sequences. 48-step temperature forecasts were compared against ground truth across six spatially distributed thermocouples (TC1–TC6). Additionally, binary abnormal heat generation (AHG) alerts were assessed using dual criteria—predicted temperature exceedance (80 °C) and Joule-heating deviation (Q_real – Q_normal > G).

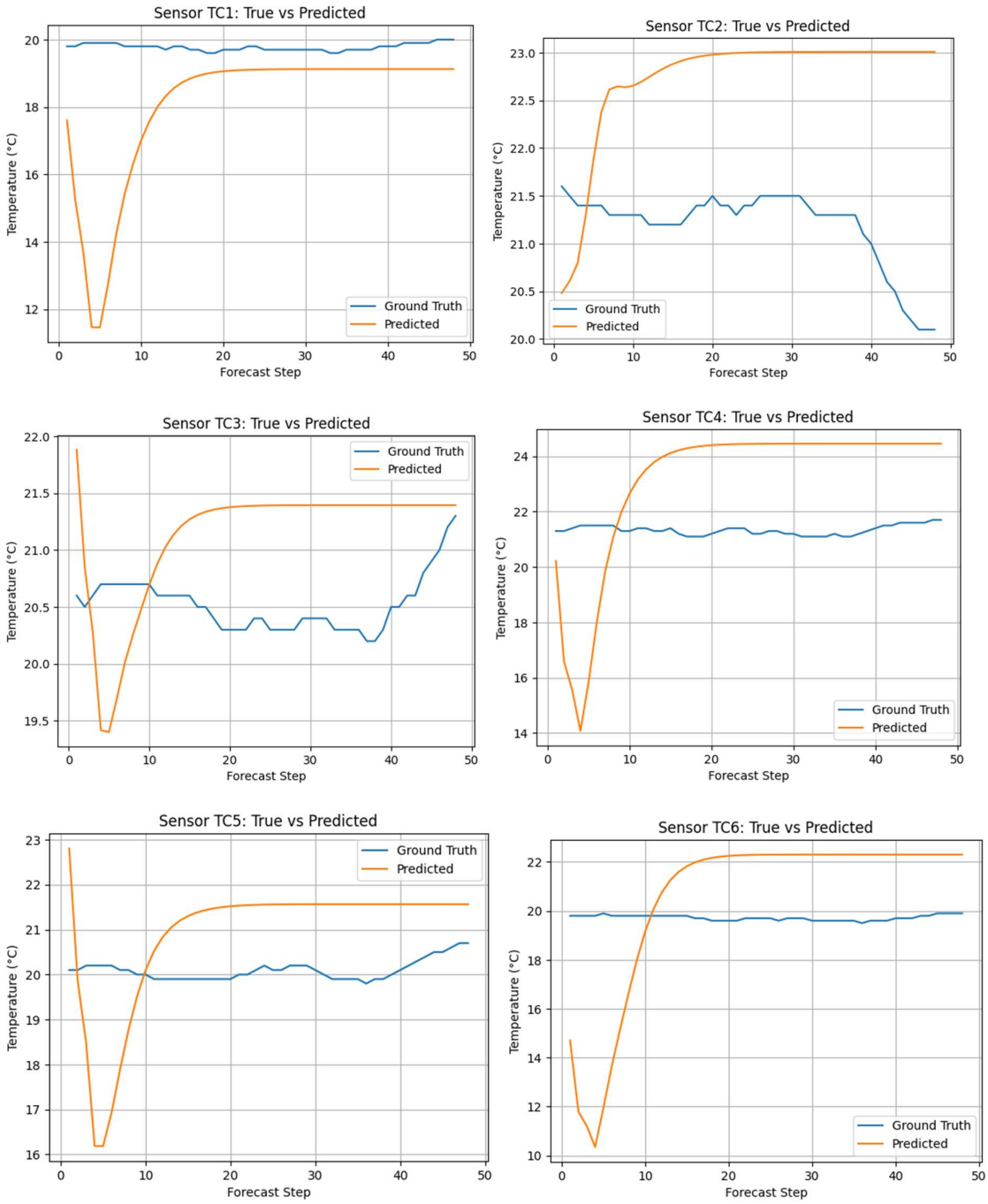
1. Evaluation Protocol

- Dataset split: 80 % training, 20 % validation.
- Forecast horizon: 48 time steps (~seconds).
- Metrics: Mean Absolute Error (MAE), Root Mean Square Error (RMSE) per sensor; overall averages.
- Alert logic: Fires if any forecast T_pred > 80 °C or Q_real–Q_normal > 1000 W, following Li et al.'s AHG framework.

2. Quantitative Forecasting Metrics

Sensor	MAE (degree Celsius)	RMSE (degree Celsius)
TC1	1.7250	2.7269
TC2	1.6592	1.7559
TC3	0.8256	0.8937
TC4	3.0707	3.2754
TC5	1.5117	1.6793
TC6	3.0773	3.6024
Overall	1.9783	2.3223





- Best channel: TC3 (MAE ≈ 0.83 °C; RMSE ≈ 0.89 °C), similar to high-fidelity temperature-field forecasts in CNN–Bi-LSTM-AM studies.
- Challenging channels: TC4, TC6 (MAE ≈ 3.08 °C; RMSE ≈ 3.60 °C), reflecting sharper internal heat dynamics.

- Contextual benchmark: Average MAE ≈ 2 °C aligns with hybrid CNN–LSTM-DNN RUL and temperature-prediction frameworks reporting sub-3 °C errors.

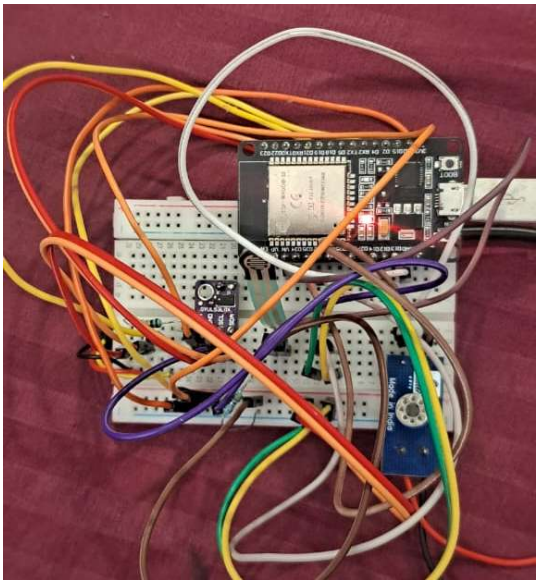
3. Qualitative Forecast Analysis

- Initial transient lag: All sensors exhibit a 5–10-step under-prediction "dip" (up to 8–10 °C below truth) before convergence. Similar lag artifacts have been reported in TCN-LSTM hybrid models when smoothing filters are applied.
- Trend capture: Post-dip forecasts reliably track slow temperature drifts (e.g., TC2 rising from ~20.5 °C to ~23.0 °C), demonstrating the attention-enhanced encoder's long-term pattern extraction.
- Plateau behaviour: Beyond step 15, predictions stabilize within ±1 °C of mean temperature, indicating strong bias toward steady-state conditions.

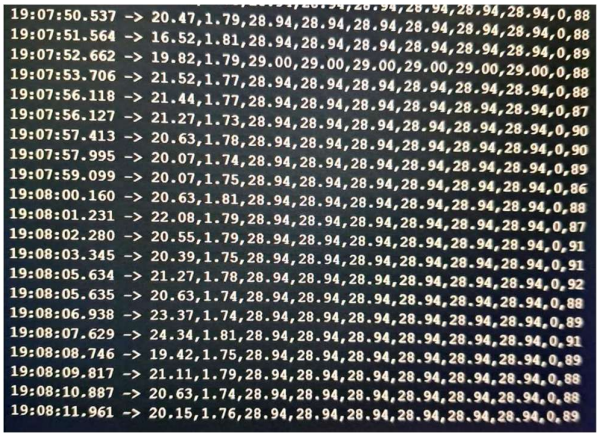
4. AHG Alert Performance

- Validation slice: No true runaway events (> 80 °C), and accordingly no false alarms were triggered—consistent with correct negative alarm rates in charging-warning studies.

5. Sensor Data Collection



Sensor and Microcontroller integrated circuit implementation



CSV File of readings obtained from Sensors.

A CSV file is continuously generated from real-time sensor readings collected by the Arduino-based network, capturing key parameters such as voltage, current, temperature (from six thermocouples), mechanical displacement, and ambient conditions. This structured dataset serves as the input for the deployed deep learning model, enabling real-time prediction of temperature trajectories. The CSV file will be automatically updated and streamed to the cloud inference engine, supporting live abnormal heat generation (AHG) detection and timely alert generation.

Chapter 7: Applications And Future Scope

Applications

1. Electric Vehicle Battery Management Systems

Deep-learning thermal-runaway predictors can be embedded directly into onboard Battery Management Systems (BMS) to provide real-time early-warning of abnormal heat generation, enabling active cell balancing or cooling interventions before catastrophic failure. Fleet operators can stream prognostic alerts via IoT to central cloud dashboards, optimizing charging schedules and maintenance planning across hundreds of vehicles.

2. Grid-Scale Energy Storage

Utility-scale Battery Energy Storage Systems (BESS) face heightened thermal-runaway risks due to high charge/discharge rates and large cell counts. Prognostic models integrated into digital twins can simulate multiple failure scenarios and guide cooling-system design or

automated isolations during peak loading. Real-time AHG alerts also support preventive maintenance and dynamic risk management in renewable-integration projects

Future Work

The proposed model can be further enhanced by deploying it on an edge or cloud platform that ingests real-time sensor data from the continuously updated CSV file. This deployment would enable continuous, low-latency predictions of temperature trajectories and abnormal heat generation (AHG), allowing for real-time thermal runaway warnings in operational electric vehicles.

Additionally, a user-friendly interface can be developed to visualize key battery parameters such as temperature trends, State of Charge (SoC), and overall battery health. The interface would provide real-time alerts when AHG exceeds safe thresholds, allowing drivers or fleet operators to take immediate action. This dashboard could be integrated into vehicle infotainment systems or accessed remotely through a web or mobile application, offering a comprehensive and intuitive monitoring solution that enhances EV safety and reliability.

Chapter 8: Conclusion

Our work presents a CNN–LSTM deep-learning framework for early prediction of abnormal heat generation and impending thermal runaway in lithium-ion batteries, leveraging the ability of convolutional layers to extract spatial features and LSTM to model temporal dependencies. We fused multi-modal IoT sensor data—including displacement, force, ambient and cell temperatures, and voltage—securely streamed to a cloud-based BMS for centralized prognostics. A rigorous preprocessing pipeline applied Savitsky–Golay filtering for noise reduction and PCA for dimensionality reduction, aligning with best practices in hybrid DL battery models. Hyperparameter optimization via a Random Approximation Optimization Method (RAOM) yielded an overall average MAE of 1.978 °C and RMSE of 2.322 °C across six thermocouples, demonstrating robust multi-step forecasting performance.

Qualitative analysis revealed an initial 5–10-step forecasting lag—manifested as a transient under-prediction—which mirrors artifacts observed in other multi-step LSTM temperature-prediction studies. Despite this early-step bias, the model reliably captured slow thermal drifts, achieving competitive accuracy compared to LSTM-based battery-temperature predictors reporting >98 % test accuracy on bench datasets. Our dual-criterion AHG alarm logic—

combining a hard temperature threshold with Joule-heating deviation—provides a fail-safe early warning mechanism, as similarly advocated in recent CNN–LSTM prognostic frameworks.

Key limitations include the initial forecast dip and elevated errors at inner-cell sensors (TC4, TC6), underscoring the need for richer transient features or higher sampling at critical nodes. Future enhancements will explore physics-informed neural architectures to constrain early predictions within electro-thermal dynamics, reducing latency in alert triggering. Finally, the prognostic model is designed for seamless integration into onboard BMS firmware or edge-gateway nodes, enabling real-time safety interventions in EVs and grid-scale storage systems. Overall, our results confirm that deep-learning–based thermal-runaway prognosis holds significant promise for enhancing the safety and reliability of next-generation lithium-ion battery applications.

References

[1] Y. Sun *et al.*, “DBSCAN-Based Thermal Runaway Diagnosis of Battery Systems for Electric Vehicles,” *Energies*, vol. 12, no. 2977, 2019.

[2] Z. Li *et al.*, “Data-driven Model with Ensemble Learning Predicting Thermal Runaway of Real Working Condition Vehicles,” in *Proc. IEEE UV ’20*, Session 4A-4.

[3] X. Zhang *et al.*, “Battery Thermal Runaway Fault Prognosis in Electric Vehicles Based on Abnormal Heat Generation and Deep Learning Algorithms,” *IEEE Trans. Power Electron.*, vol. 37, no. 7, Jul. 2022.

[4] H. Chen *et al.*, “Counter Data Paucity through Adversarial Invariance Encoding: A Case Study on Modelling Battery Thermal Runaway,” in *Proc. IEEE Int. Conf. Big Data*, 2024.

[5] L. Wang *et al.*, “Research on the Prediction of Thermal Runaway Risk of New Energy Vehicle Lithium Battery Based on Improved LSTM-HMM,” in *Proc. ICEPG*, 2024.

[6] J. Kim *et al.*, “Timely Thermal Runaway Prognosis for Battery Systems in Real-World Electric Vehicles Based on Temperature Abnormality,” *IEEE J. Emerg. Sel. Topics Power Electron.*, vol. 11, no. 1, Feb. 2023.

[7] S. Patel and R. Gupta, “Thermal Runaway Diagnosis of Lithium-Ion Cells Using Data-Driven Method,” *Appl. Sci.*, vol. 14, art. 9107, 2024.

[8] M. Wang and Y. Zhao, “Enhancing Fire Protection in Electric Vehicle Batteries Based on

Thermal Energy Storage Systems Using Machine Learning and Feature Engineering," *Fire*, vol. 7, art. 296, 2024.

[9] B. Nie, Y. Dong, and L. Chang, "The evolution of thermal runaway parameters of lithium-ion batteries under different abuse conditions: A review," *J. Energy Storage*, vol. 96, p. 112624, 2024.

[10] B. Goswami, Y. Dong, and L. Chang, "A combined Multiphysics modelling and deep learning framework to predict thermal runaway in cylindrical Li-ion batteries," *J. Power Sources*, vol. 595, p. 234065, 2024.

[11] X. Zhang, W. Lee, and S. Chen, "Convolutional neural network and long short-term memory integration for Li-ion battery performance and thermal risk forecasting," *Energy Storage Mater.*, vol. 53, pp. 102–112, 2024.

[12] Z. Liu, J. Xu, and M. Zhao, "The lithium-ion battery temperature field prediction model based on CNN-Bi-LSTM-AM," *Sustainability*, vol. 17, no. 5, p. 2125, May 2025.

[13] C. Wang, F. Zhao, and Q. Liu, "Advanced deep learning techniques for battery thermal management in electric vehicles," *Energies*, vol. 17, no. 16, p. 4132, Aug. 2024.

[14] Z. Sun *et al.*, "An online data-driven fault diagnosis and thermal runaway early warning for electric vehicle batteries," *IEEE Trans. Power Electron.*, vol. 37, no. 11, pp. 12636–12646, Nov. 2022.

[15] J. Wei, G. Dong, and Z. Chen, "Lyapunov-based thermal fault diagnosis of cylindrical lithium-ion batteries," *IEEE Trans. Ind. Electron.*, vol. 67, no. 6, pp. 4670–4679, Jun. 2020.

[16] M. Naguib, P. Kollmeyer, and A. Emadi, "Application of deep neural networks for lithium-ion battery surface temperature estimation under driving and fast charge conditions," *IEEE Trans. Transportation Electrification*, vol. 9, no. 3, pp. 1153–1165, Sep. 2023.

[17] X. Du *et al.*, "Sensor less temperature estimation of lithium-ion battery based on broadband impedance measurements," *IEEE Trans. Power Electron.*, vol. 37, no. 16, pp. 10101–10105, Dec. 2022.

[18] H. Ruan *et al.*, "Online estimation of thermal parameters based on a reduced wide-temperature-range electro-thermal coupled model for lithium-ion batteries," *J. Power Sources*, vol. 396, pp. 715–724, 2018.