



OPEN An intelligent fault detection (IFD) system for lithium-ion battery using machine learning approach

Rahul Kumar Kamboj^{1✉}, Mukesh Singh¹, Ashima Singh² & Anju Bala²

In recent years, electric vehicles (EVs) have become increasingly popular, driven by advancements in battery technology, growing environmental awareness, and the demand for sustainable transportation. Compared to internal combustion engines, EVs not only produce fewer emissions but also offer greater energy efficiency, leading to reduced operating costs. Despite these advantages, concerns about battery failures have been a significant safety issue for EVs. This paper introduces an Intelligent Fault Detection (IFD) system—a proactive approach that utilises advanced intelligent techniques for detecting faults in EVs batteries. This paper involves developing and implementing an ML-based fault detection mechanism to monitor and safeguard the batteries from various faults, including thermal protection, under-voltage, and over-voltage. This research involves real-time data sourced from sensors embedded within the battery, which ensures the continuous collection of battery data during the vehicle's operation. The pre-processed data is analysed using a K-means clustering algorithm to classify essential groups, helping to define a valid range for temperature and voltage. Further, the ensemble approach has been used to classify safe and unsafe areas for battery operations. The proposed model has 0.94 accuracy for identifying faults, which contributes to the long-term sustainability and economic viability of electric mobility.

Keywords Battery management system, Ensemble learning, K-means clustering, Intelligent fault detection, Safe operating area

The use of electric vehicles has been rapidly increasing due to their environmental benefits and advancements in battery technology. However, one of the significant challenges faced by EVs is to optimise battery performance and their operational life extension¹. As the demand for EVs is increasing in our daily lives, the protection issues are also increasing. So, there is a need for a fault detection system that can detect these faults intelligently and proactively to anticipate issues before they escalate. Such fault detection systems improve not only the safety and dependability of EVs, but also promote the overall sustainability². A Battery Management System (BMS) is a method used to control and regulate power during the battery's charging and discharging phases. It safeguards the cells by observing battery capacity, computing subsidiary data, and helping to balance the battery properly³. The BMS identifies voltage, temperature, or current abnormalities, prompting the alarm system to protect the battery pack. These abnormalities are caused by the faults that occur in the battery. Different types of faults can occur during battery operation, including overcharging, over-discharging, and thermal issues⁴. Overcharging is a fault that can occur in a battery when it is charged beyond its recommended voltage limit.

This can happen for various reasons, such as charging the battery beyond the manufacturer's specified voltage, faulty charging equipment, and extended charging time. In extreme cases, this may increase the temperature of the battery and the battery can catch fire or explode. Hence, it is essential to monitor the charging process carefully and to use reliable chargers to avoid overcharging⁵. A fault in the battery can also occur due to over discharging, which happens when the battery is discharged beyond its recommended voltage or capacity limits. This can be caused by prolonged usage without recharging, improper storage, or a faulty battery management system. Over discharging can lead to reduced capacity and performance. It is essential to monitor and prevent over-discharging to ensure the longevity and safety of the battery. Moreover, thermal fault in a battery occurs when there is an abnormal increase in temperature, leading to potential damage or failure⁶. Several reasons contribute to thermal faults in batteries, such as overcharging, over-discharging, external heat sources, manufacturing defects, and mechanical damage. Monitoring and controlling the temperature during

¹Electrical and Instrumentation Engineering, Thapar Institute of Engineering and Technology, Street, Patiala, Punjab 147004, India. ²Computer Science and Engineering, Thapar Institute of Engineering and Technology, Street, Patiala, Punjab 147004, India. ✉email: rkamboj_phd22@thapar.edu

battery operation is essential to prevent thermal faults. Proper cooling systems, temperature sensors, and battery management systems are crucial in maintaining safe operating conditions and preventing thermal incidents.

There are various types of electrical faults that can occur in lithium-ion batteries, as referred to in Table 1, and they differ in how easily they can be detected. Faults such as overcharging and over-discharging are generally easy to identify, as they produce clear and measurable changes in the battery’s behaviour. In contrast, faults like short circuits, voltage imbalances, and (under-over) temperature due to excessive heat or cold are more difficult to detect. These types of faults often develop gradually or arise under specific conditions, making real-time identification challenging. This underscores the importance of the IFD system to detect such hidden faults early, ensuring the safety, performance, and longevity of the battery.

An IFD system for EV batteries is designed to monitor and analyse battery behaviour in real time to detect potential issues before they lead to serious failures. By using advanced machine learning algorithms and data collected from onboard sensors—such as voltage, current, and temperature the system can identify unusual patterns that indicate early signs of battery faults. These intelligent models learn from both historical and live data, allowing them to differentiate between normal battery fluctuations and actual problems, such as cell imbalance, thermal anomalies, or capacity degradation. This proactive approach not only improves safety and extends battery life but also supports predictive maintenance by alerting users or service systems before a fault becomes critical. As EV adoption grows, such smart battery monitoring systems are becoming essential for ensuring reliable performance, reducing downtime, and building trust in electric mobility.

Literature survey

Various researchers are working on fault detection of batteries, like Abada et al.⁷, who discussed the hardware safety approach of batteries, mainly focused on the modelling and testing of batteries under different conditions. The modelling of the battery provided a thermal runaway for a pack to integrate the ageing of the battery effect, which helps to assess the battery functionality and desired safety aspects of the battery.

Similarly, Jiang et al.⁸ provided a hardware system for future safe direction for charging systems by comparing onboard connected batteries, cell energy density and efficiency with the long life cycles. Meissner et al.⁹ presented battery monitoring and energy management, which predicted the future power of batteries by generating a data set for complete charging periodically with well-defined SOC and the data generation on the battery with different test benches. Chen et al.¹⁰ employed the RC equivalent circuit model hardware setup and a swarm optimisation technique to ascertain battery parameters. Additionally, a two-layer model was utilised based on an external short circuit and a fault diagnosis approach to estimate defects in a battery. Similarly, Wang et al.¹¹ put forth a modified Shannon entropy based on cell voltage data. A security management technique was also provided, which used Z-score normalisation. Zhu et al.¹² proposed an ML approach for determining the temperature of a battery at a specific time to mitigate the risk of thermal runaway, along with the decomposition of temperature with reversible and irreversible heat with battery operation components. However, the voltage parameter was missing in this research. Chen et al.¹³ proposed a method for voltage problem identification in lithium-ion batteries utilising the Local Outlier Factor algorithm. The LOF was utilised to measure the degree of deviation of parameters from their neighbouring values. Consequently, faults were identified and evaluated using an outlier filter based on the Grubbs criterion^{14,15}. Additionally, Zhao et al.¹⁶ carried out a study combining simulation and experimental analysis of external and internal short circuits. A modified electrochemical-thermal coupling model was used to predict temperature variations; however, it did not indicate active fault status.

Feng et al.¹⁷ investigated fault identification in large-format lithium batteries, specifically focusing on internal short-circuit faults. This paper aimed to utilise voltage and temperature responses to develop an algorithm capable of identifying the occurrence of faults and determining their particular position within the battery. Moreover, studies have been conducted on the modelling and analysis of battery short circuits; for example, Adnan et al.¹⁸ introduced a methodology that utilises a support vector machine (SVM) to include the diagnosis and prognostics of battery health. Similarly, Zhang et al.¹⁹ introduced a novel approach to the diagnosis of capacity problems online and in real time in parallel connected lithium ionhium ionhium ion battery groups. For a better approach, there is a need for more accurate diagnostic methods.

Category	Fault type	Easy to detect	Hard to detect
Electrical	Short circuit	✓	✗
	Internal short	✗	✓
	Capacity fade	✗	✓
	Self-discharge	✗	✓
	Under voltage	✗	✓
	Over voltage	✗	✓
Thermal	Over-temperature	✓	✗
	Thermal runaway	✗	✓
	Under-temperature	✓	✗
Mechanical	Sensor fault	✓	✗
	Physical damage	✗	✓

Table 1. Classification of battery faults.

Further, the outlier analysis is a technique employed in fault diagnostics^{20–22}. However, the majority of traditional statistical analyses conducted on electric vehicles are unable to identify the specific defective battery cells effectively. In addition, the currently employed approaches are incapable of identifying the emergence of faults because the voltages remain within the acceptable range. To overcome this, Hu et al.²³ introduced a technique for estimating battery capacity using a combination of particle swarm optimisation and k-nearest neighbour regression. Yang et al.²⁴ utilised an extreme learning machine (ELM) based thermal model to represent the thermal response of batteries during external short circuits. However, the aforementioned models possess limited applicability since they are solely suitable for diagnosing specific defects within specific operational circumstances. Fault detection was performed by comparing parameter values against predefined criteria. While these methods are easy to implement, they require multiple thresholds, posing challenges in accurately determining their values²⁵. Furthermore, AI techniques, such as LS-SVM and LSTM neural networks, have been utilised for diagnosing issues in LIBs²⁶. However, it is important to acknowledge that these methods rely on a large volume of historical data, which can be challenging to access and acquire.

Yukseket al. emphasise the need for accurate SOH estimation to support efficient battery management. While traditional methods require large datasets and heavy processing, they propose a simple curve-fitting approach that delivers high accuracy with minimal data. Remarkably, using just 1% of the Oxford battery dataset, their model outperformed other methods based on RMSE, showing strong results with low cost and complexity²⁷.

Lale shows that lithium-based batteries, while powerful, are sensitive to heat and charging speed. To tackle this, they propose an adaptive BMS that adjusts charging rates based on temperature. This smart approach boosts efficiency, extends battery life, and in simulations, improved energy output by 11.49% compared to standard methods, proving the value of temperature-aware charging²⁸.

Yamacli highlights that battery safety and ageing are key challenges, especially in electric vehicles. Unlike most methods that focus on single cells, this study looks at series-connected batteries in real-world setups. Using deep learning with hybrid classification, and data from Oxford and CALCE, the model achieved 98.33% accuracy with low error, proving it's both accurate and practical for real-time SOH monitoring²⁹.

Shang et al. point out that as we move toward carbon neutrality, keeping lithium-ion batteries safe and reliable is crucial. They review three main ways to detect battery faults: data-driven, model-based, and threshold-based, each with its pros and cons. Their work offers a clear summary of fault types and the latest techniques, helping guide future research in better battery fault detection³⁰.

The literature has discovered that few authors have used machine learning approaches for over-temperature, under-temperature, over-voltage and under-voltage protection. Consequently, there is a need for an intelligent fault detection (IFD) system that continuously oversees the battery's status and establishes protective zones to prevent potential issues. This system alerts the user upon detecting hazardous conditions, enabling timely intervention to safeguard the battery from significant faults. This method aims to shield battery cells from potential harm and optimise safe operational ranges for batteries.

Contribution

The major contributions of this research are as follows:

- This approach uses real-world test data from LiFePO₄ batteries to analyse fault patterns and inconsistencies within the battery pack under various operating conditions.
- For better optimisation, the study utilises two test benches for data generation, facilitating the identification of aberrant battery conditions.
- An IFD has been proposed to detect the faults in EVs proactively to improve reliability.
- By factoring in temperature and voltage, this model significantly reduces inaccuracies and enhances performance, resulting in shorter computational times while improving the accuracy of the estimation.

Novelty

The need for a fault detection approach of lithium-ion batteries defines the limits of voltage and temperature within which the battery can function safely. Recent research focuses on developing a novel approach that adjusts in real time based on battery health, usage patterns, and environmental conditions. By integrating machine learning, dynamic frameworks improve safety, extend battery life, and enable higher performance. This shift from static to data-driven for the fault detection approach represents a key innovation for applications such as electric vehicles and energy storage systems.

Organisation

“Proposed model intelligent fault detection system (IFD)” section discusses the proposed model with data generation and data preprocessing. The fault detection model is represented in “Intelligent fault detection model” section for safe operating areas. “Results and discussion” section discusses the results and discussion, and “Proposed model intelligent fault detection system (IFD)” section concludes the research findings.

Proposed model intelligent fault detection system (IFD)

The main goal of this research study is to propose a fault detection model as depicted in Fig. 1. The study comprises three main phases. Initially, data acquisition takes place where real-time data is gathered from two test benches to form a dataset. In the second phase, the data are preprocessed to remove null values, and the K-means clustering algorithm is commonly used to categorise temperature and voltage ranges. Following this, the pre-processed data is applied in a machine learning technique, which is an ensemble model that identifies both secure and vulnerable battery regions. The proposed system uses an ensemble machine learning model to estimate the safe operating area. The third phase provides a comprehensive analysis of the functionality and

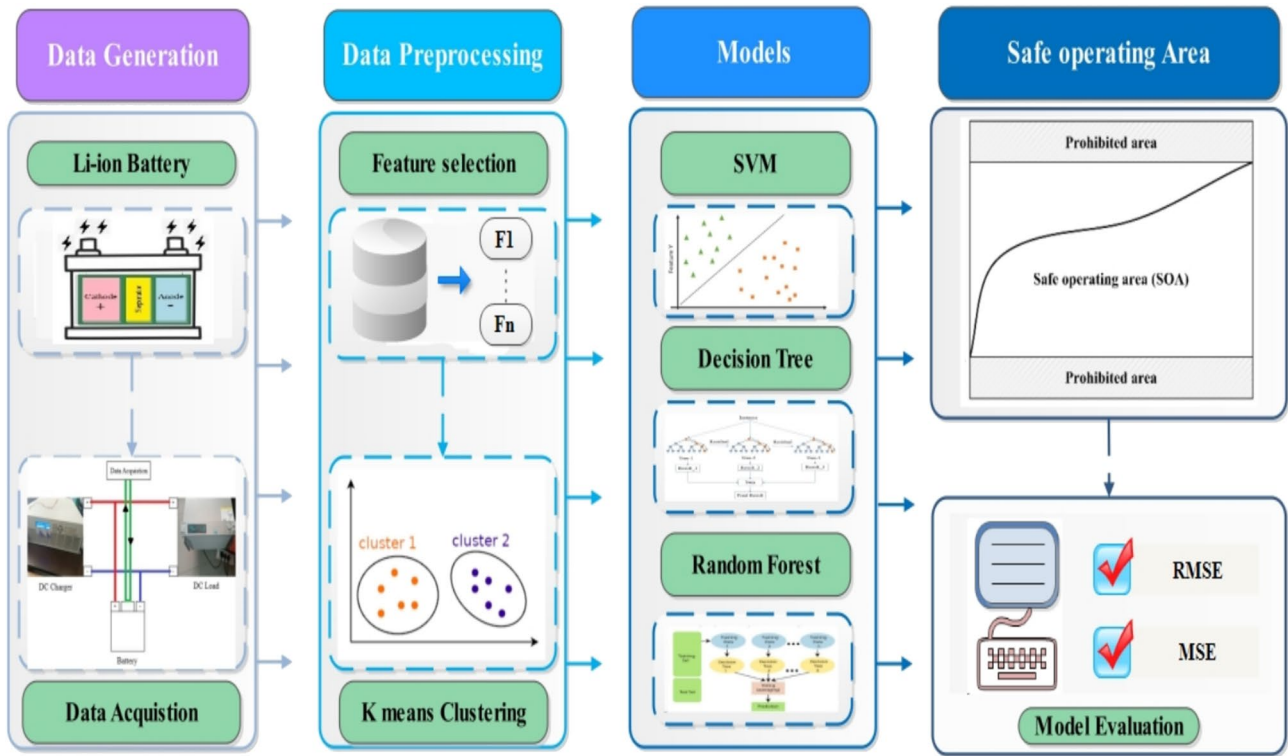


Fig. 1. Proposed IFD model.

S.no.	Battery parameter	Value
1	Manufacturer	Likraft
2	Nominal capacity	100 Ah
3	Nominal voltage	51.8 V
4	Recommend charging current	25 A
5	Cycle life	2000 Times
6	Operating temperature	5–65°C
BMS specification		
1	Max discharge current	100 A
2	Max charge current	50 A
3	Voltage	48–60 V
4	Communication	WIFI, Bluetooth
5	Software	Intellicar

Table 2. Characteristics of a LiFePO4 battery pack.

unique attributes of each model operating on real-time data. This model decision is critical to safeguarding the battery against potential harm caused by extreme temperature or voltage levels. Monitoring the battery’s performance is essential to protect it from potential damage and extend its lifespan.

Data acquisition

The dynamic recording of input parameters, such as voltage and temperature, is conducted on test benches to collect real-time data during the charging and discharging operations. The study used a LiFePO4 battery pack with a nominal capacity of 100 Ah and a voltage rating of 51.2 V for data generation. The battery’s characteristics and BMS specifications are delineated in Table 2. The battery holds a smart BMS, with an instrument cluster and computer-controlled software, called Intellicar Telematics.

Test bench

There are two types of test benches used to generate data for better accuracy in fault detection. Test bench 1 operates under a constant charging and discharging load, while test bench 2 utilises Ekart to collect data. Fig.

2 represents the test bench as a load. The datasets produced by these test benches are labelled as DS1, FD and DS2, respectively.

- In Test Bench 1, a controlled setup is designed to simulate real-world battery charging and discharging conditions. A DC power supply is used to charge the battery, while a DC electronic load facilitates discharging, both operating under a constant current mode to ensure consistency during experiments³¹. The overall configuration of this test bench is shown in Fig. 2. Specifically, a 15 kW programmable power source from Keysight is used for battery charging, whereas a 15 kW electronic load from Chroma is employed to manage the discharging process. To monitor thermal behaviour during cycling, a temperature sensor integrated with a smart BMS tracks the temperature of individual cells. The test procedure involves repeated charging and discharging cycles at a current of 25 A to generate a rich and diverse dataset. The charging process follows the Constant Current–Constant Voltage (CC–CV) protocol, continuing until the battery voltage reaches 55 V, corresponding to 100% SOC. After a short resting phase, the battery is discharged at a constant current of 0.25C until it hits the predefined cutoff voltage, simulating typical use scenarios.
- The faulty data (FD) were intentionally generated to rigorously test the model's diagnostic accuracy under abnormal and potentially hazardous conditions. These faults were not incidental but were deliberately engineered within the test setup to replicate realistic degradation and failure scenarios observed in practical battery systems. The faults were introduced during various phases of the battery cycle by applying controlled disturbances, such as abrupt voltage fluctuations induced by modulating the power source output during the charging phase, and temperature spikes simulated through the application of an external heat source to specific cells, mimicking thermal runaway or localised overheating. Additionally, current surges were introduced to simulate short circuits or load anomalies, and sensor disconnections or drift were simulated to test the system's resilience against data quality degradation and partial monitoring failures. All faults were introduced incrementally and under tight supervision to ensure safety and preserve the integrity of the hardware. This behaviour replicates real-world safety protocols where systems are required to enter protective states upon detecting anomalous behaviour. To support effective training and evaluation of the model, the dataset

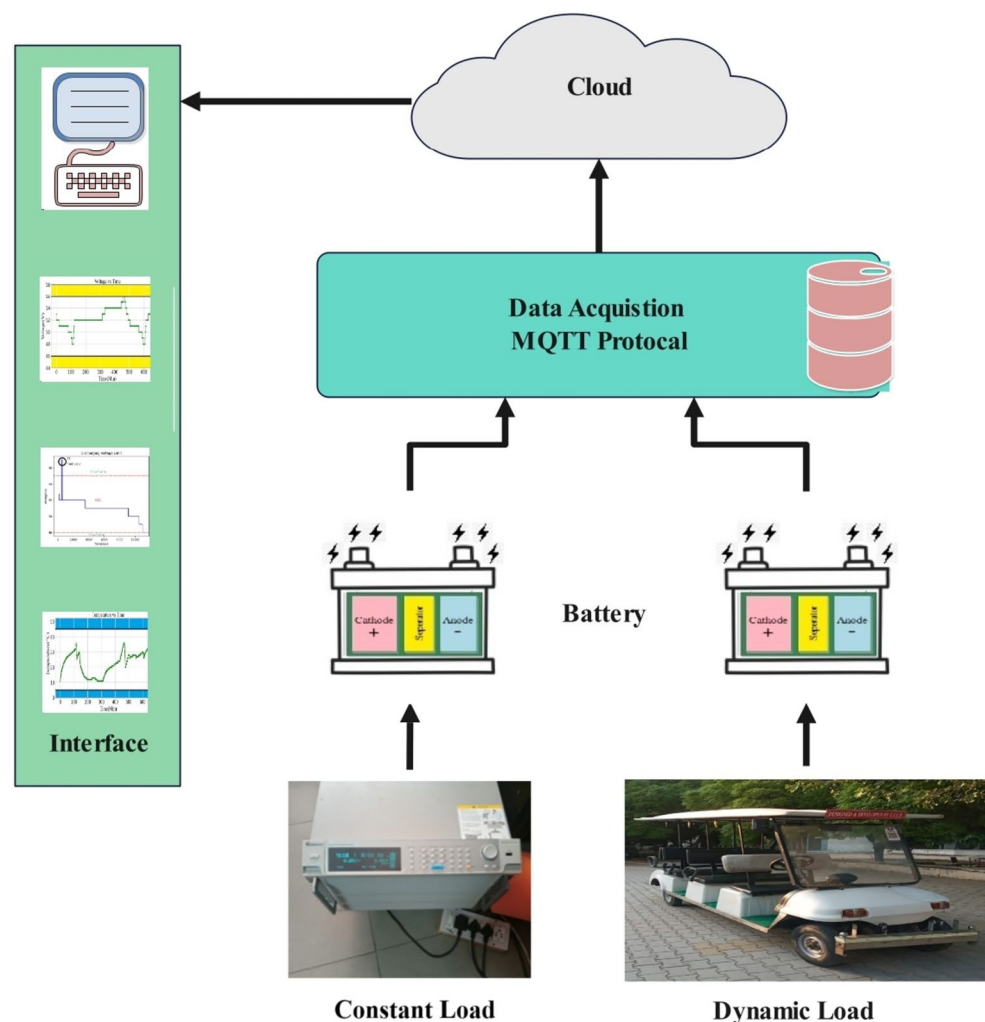


Fig. 2. Data acquisition using different load.

was designed to maintain class balance between normal and faulty instances. A median voltage reference of 51.5 V was used to categorise the samples. Similarly, for temperature, 33.5 °C was set as the neutral boundary, with higher values flagged as thermally abnormal. This labelling strategy aids the learning algorithm in distinguishing subtle patterns associated with different fault types. Moreover, the dataset captures not just steady-state anomalies but also transient behaviours, such as short-lived spikes and gradual drifts, enhancing the model's ability to detect both abrupt and progressive failures. All data were collected using high-precision instruments and synchronised time-logging, ensuring a high signal-to-noise ratio and enabling robust benchmarking. This meticulously curated faulty dataset thus serves as a vital component in evaluating and validating the model's capacity to detect, classify, and respond to early signs of battery degradation and malfunction, offering strong potential for real-world fault diagnosis in battery-powered systems.

- In Test Bench 2, a custom-designed E-kart is used to discharge the battery under real-world driving conditions, allowing for the collection of dynamic and randomly varying data samples. This experimental setup is intended to evaluate the system's performance in practical scenarios. A 1300-watt charger is employed to recharge the battery between runs, ensuring it is ready for repeated testing.

The E-kart is powered by a 2500-watt motor, which can draw a peak current of up to 100 A. This high load creates variable and realistic discharge profiles that reflect diverse driving behaviours, including acceleration, deceleration, and stop-and-go traffic. These fluctuations naturally generate a rich dataset, capturing the complexity of actual vehicle operation. The data collected from this test setup is referred to as the DS2 dataset. It includes parameters continuously monitored and recorded by the BMS, such as voltage, current, temperature, and state of charge. Because the E-kart was designed and developed in-house, it allowed the researchers to test under controlled yet realistic driving patterns and environmental conditions. This hands-on approach ensures that the data mirrors practical battery usage and provides a solid foundation for evaluating and training the IFD model.

Table 3 presents an overview of the threshold values and corresponding sample sizes used in the development of the IFD model. These thresholds serve as critical reference points for identifying deviations in battery behaviour that may indicate potential faults. It's important to note that the threshold values are not fixed; they vary depending on the specific chemistry of the lithium-ion battery being used. For instance, batteries based on LiFePO₄, NMC, or LCO chemistries may each exhibit different safe operating limits for parameters like voltage and temperature.

The dataset used for model training and evaluation includes samples collected across a range of operating conditions, reflecting both normal and faulty states. Each sample is labelled and aligned with its corresponding threshold to help the IFD model learn to distinguish between healthy and abnormal behaviour accurately. By tailoring the thresholds to the battery's chemistry and capturing sufficient data across various states, the model becomes more adaptable and reliable in real-world applications.

Data processing

The raw data collected from the battery is further preprocessed. The data pre-processing plays a vital role in data analysis and machine learning methodologies, as it converts raw input data into a suitable format for further analysis. This process involves cleaning and feature selection to enhance its quality and manageability³². Cleaning involves addressing duplicates and missing values, with the removal of duplicates being necessary to avoid biased analysis. Whereas, feature selection is essential for improving model performance, reducing computational complexity, and avoiding over-fitting. Overall, data pre-processing is essential for ensuring data quality and suitability for analysis, reducing biases, noise, and inconsistencies to improve output accuracy. Subsequently, the preprocessed dataset undergoes additional processing with K-means clustering to identify valid data ranges.

K means clustering

Clustering is used to classify input data into groups, segmenting a data set into groups based on shared characteristics³³. When categorising data such as voltage and temperature ranges, the initial choice of cluster centres plays a crucial role in determining the quality of the resulting clusters, as the algorithm can potentially converge to a local minimum. To address this concern, K-means is often run multiple times with varying initialisations, and the optimal outcome is chosen. During the algorithm's update stage, outliers are heavily influenced by cluster mean calculations and potentially impact the clustering results³⁴. To partition dataset {*d*₁, *d*₂, . . . , *d*_{*n*}} into clusters {*K*₁, *K*₂, . . . , *K*_{*k*}}, each containing a subset of data points. The goal is to find centres {*C*₁, *C*₂, . . . , *C*_{*k*}} that minimise the sum of squared distances between each data point and its respective cluster centre. This objective function is expressed mathematically as:

Data	Time	Sample size	Threshold for voltage	Threshold for temperature
DS1	600 min	7200	46–56 V	5–45 °C
DS2	600 min	7200	46–56 V	5–45 °C
FD	200 min	2400	46–56 V	5–45 °C

Table 3. Specification for LiFePO₄ generated dataset.

$$J(C, d_i) = \sum_{i=1}^N \sum_{j=1}^K \|d_i - C_j\|^2. \quad (1)$$

Then, the mean of the cluster is updated until the convergence criteria are met. The pseudocode for K-means is illustrated in Algorithm 1.

```

1: Input: Training dataset with features  $d_1$  (Temperature),  $d_2$  (Voltage), labeled
   with respect to  $d_3$  (Time)  $\{d_1, d_2, \dots, d_n\}$ 
2:  $K$ : Number of clusters to be formed.
3: Procedure: K-Means Algorithm
4: Initialize  $K$  centroids:  $C_1, C_2, \dots, C_K$  ▷ Random initialization
5: for each item  $d_i \in D$  do
6:   Assign  $d_i$  to the cluster with the closest centroid:
7:    $C_i \leftarrow \arg \min_j \|d_i - C_j\|^2$  ▷ Euclidean distance
8: end for
9: end for
10: for each cluster  $j$  do
11:   Update the centroid  $C_j$ :
12:    $C_j \leftarrow \frac{1}{|C_j|} \sum_{d_i \in C_j} d_i$  ▷ Mean of cluster
13: end for
14: end for
15: Convergence criteria are met
16: Output: Final cluster assignments

```

Algorithm 1. Pseudo code for K-means clustering.

Intelligent fault detection model

This section explains the proposed model using ensemble learning, which is a group of base models to foresee a fault detection system for lithium-ion batteries using Support Vector Machine (SVM), Decision Tree (DT) and Random Forest (RF) models. The output of the proposed model is labelled as the safe operating area and shows the decision boundary for the features and status of the fault.

Ensemble learning

The ensemble approach is a strategy that amalgamates multiple classification algorithms to enhance and optimise overall performance³⁵. There are two prevalent ensemble-based methods: bagging and boosting. Boosting is a sequential process where each subsequent model corrects the errors of the preceding one, while bagging consolidates the results of multiple models to derive the outcome. The proposed framework employs a voting mechanism to combine the models from the present research into an ensemble. This approach implements an ensemble approach using the majority voting method to combine the best-performing models identified in the previous training phase.

The ensemble comprises two datasets, DS1, DS2, and FD, validated through time-series division. The algorithm allocates 80% of each dataset for model training and reserves 20% for testing. The final prediction is determined by the highest number of votes among the models. The ideal model output, depicted in Fig. 3, delineates real-time battery operation, defining safe and unsafe zones based on the decision boundary in the feature space. This boundary is learned during training and used for classification during prediction.

Support vector machine

SVM is a set of supervised learning algorithms for classification that approximates complex multivariate functions with high accuracy. Initially designed for classification tasks, SVM's primary goal is to predict outcomes for data points within known data limits accurately. This is achieved through training the SVM model on a specific dataset referred to as the training set³⁶.

$$y = mx + c. \quad (2)$$

Here, m is the slope, and c is the constant. The hyperplane exists in either two dimensions or three dimensions. The SVM algorithm operates on the basis of the maximum-margin classifier, which aims to maximise the geometric margin while minimising the error value. The pseudocode for SVM is given in Algorithm 2.

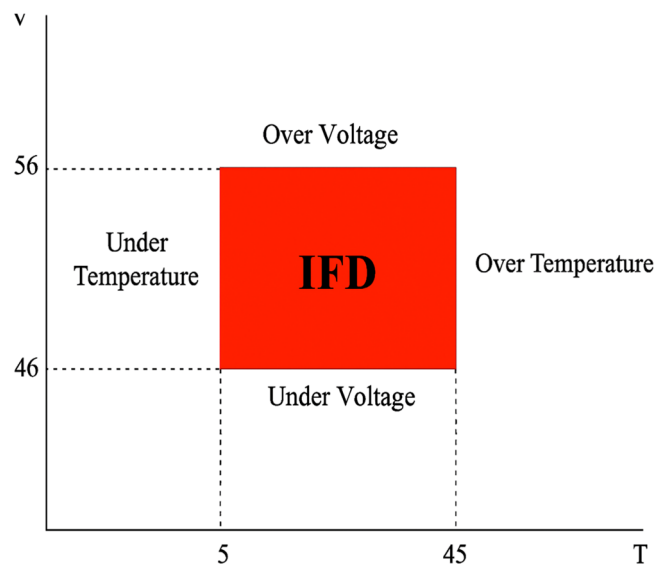


Fig. 3. IFD output.

-
- 1: **Input:** Training dataset $D = \{(v_i, t_i, y_i)\}$ Voltage v_i , Temperature t_i , Label y_i (safe or unsafe)
 - 2: **Output:** SVM model for fault estimation
 - 3: **Procedure:**
 - 4: **Train SVM Model:**
 - 5: **for** Initialize SVM parameters **do**
 - 6: Train SVM using the training dataset D
 - 7: **end for**
 - 8: **end for**
 - 9: **Test SVM Model:**
 - 10: **for** New data point $(v_{\text{new}}, t_{\text{new}})$ **do**
 - 11: Normalize and scale v_{new} and t_{new}
 - 12: Predict the fault estimation using the trained SVM model: y_{pred}
 - 13: **end for**
 - 14: **end for**
 - 15: **Output:** Predicted fault estimation y_i
-

Algorithm 2. Pseudo code for SVM.

Decision tree

The DT algorithm is a popular supervised learning method specifically developed to handle regression and classification problems³⁷. Within the framework of a DT, nodes represent features, while edges indicate the corresponding outcomes. The attribute with the greatest information gain value is chosen as the root node. This technique is repeatedly applied to each feature until the final node no longer undergoes additional partitioning. The subsequent equation determines the calculation of entropy.

The equation is given by:

$$H(S) = \sum_{c \in C} -p(c) \log_2 p(c). \quad (3)$$

Here, $H(S)$ represents entropy, where C represents the set of classes, S represents voltage and temperature in the data, and $p(c)$ represents the probability of C with respect to S . The pseudo-code for DT is represented in algorithm 3.

```

1: Input: Input Voltage  $v_i$ , Temperature  $t_i$ , Label  $y_i$  (safe or unsafe)
2: Output: DT model for fault estimation
3: Procedure: Split the data set
4: if  $V \geq$  Threshold Voltage and  $T \geq$  Threshold Temperature then
5:   return "Unsafe: High Voltage, High Temperature"
6: else if  $V \leq$  Lower Voltage and  $T \geq$  Threshold Temperature then
7:   return "Unsafe: Low Voltage, High Temperature"
8: else if  $V \geq$  Threshold Voltage and  $T \leq$  Low Temperature then
9:   return "Unsafe: High Voltage, Low Temperature"
10: else if  $V \leq$  Lower Voltage and  $T \leq$  Low Temperature then
11:   return "Unsafe: Low Voltage, Low Temperature"
12: else
13:   return "Safe Condition"
14: end if
15: end if
16: Output: Predict the fault estimation using the trained DT model.

```

Algorithm 3. Pseudo code for DT.

Random forest

RF builds multiple decision trees using different subsets of the dataset and aggregates their predictions to determine the final result. Essentially, it is an ensemble of simple decision trees. After combining the outputs from each tree, a voting process takes place, where the classification outcome is decided based on the majority vote. Algorithm 4 presents a concise pseudo-code explanation of how RF operates. The thermal runaway threshold indicates the highest and lowest temperatures of the battery, safeguarding it against thermal runaway and optimising performance using real-time data. Once it comes to voltage, the classifier determines the range based on charging and discharging voltage, while for temperature, it specifies only the min-max range³⁸. The decision for RF depends on the gain of entropy.

Require:

```

1: Input: Training dataset  $D = \{(v_i, t_i, y_i)\}$  where  $v_i$  is voltage,  $t_i$  is temperature,
   and  $y_i$  is the output label.
2: Output:  $RF$  fault detection.
3: Procedure:
4: Split  $D$  into training set  $D_{\text{train}}$  and validation set  $D_{\text{val}}$ .
5: for each decision tree  $T$  in Random Forest do
6:   Randomly select a subset  $S_i$  from  $D_{\text{train}}$ .
7:   Train decision tree  $T$  using  $S_i$ .
8: end for
9: end for
10: Function Classification( $v, t$ ):
11: Initialise predictions as a new list.
12: for each decision tree  $T$  in  $RF$  do
13:   Add the class predicted by  $T$  for  $(v, t)$ .
14: end for
15: end for
16: Compute the final output as the average or majority vote of the predictions.
17: Train Random Forest  $RF$  using  $D_{\text{train}}$ .
18: Output: Predict the fault estimation using the trained RF model.

```

Algorithm 4. Pseudo code for RF.

Information gain represents the anticipated decrease in entropy resulting from dividing the examples based on a specific attribute. A greater information gain indicates that the attribute is more successful in categorising the training data. The final prediction results from a majority vote. This approach employs multiple decision trees through ensemble learning to improve the accuracy and robustness of the classifier³⁹.

Hyperparameter tuning using Bayesian optimisation

Bayesian Optimisation offers a smart and efficient way to tune ensemble models used in fault detection, especially when working with complex and high-dimensional data, like that collected from electric vehicles. In

ensemble techniques such as Random Forest (RF), Support Vector Machine (SVM), and Decision Tree (DT), the performance of the model depends heavily on the choice of hyperparameters. Rather than testing every possible combination, Bayesian Optimisation builds a predictive model that estimates how different hyperparameter settings might affect key performance metrics, like accuracy, precision, recall, or F1-score. It then uses this model to choose the most promising hyperparameter combinations, balancing the need to explore new options with refining those that already show good results. This approach helps save time and computational resources while improving the model's effectiveness. In the context of fault detection, where anomalies are often rare, inconsistent, or difficult to distinguish, Bayesian Optimisation plays a crucial role in fine-tuning ensemble models to reliably differentiate between normal and faulty conditions. The hyperparameters are shown in Table 4. This makes it especially useful when traditional trial-and-error tuning or exhaustive searches would be too time-consuming or inefficient.

In this proposed model approach, individual models are trained on historical temperature and voltage data, labelled with the safe operating area. The outputs of the three classifiers are then aggregated using majority voting or weighted averaging to derive the final prediction. This combination leverages the diversity of the base learners to reduce the overall prediction error and improve sensitivity to various fault patterns. The ensemble model is particularly suitable for real-time monitoring applications, where early and accurate fault detection is crucial for system stability and preventive maintenance. The pseudo-code for the ensemble is represented in Algorithm 5.

Require: ML Models: SVM, DT, RF; Dataset X , labels Y

Ensure: Higher Voted Models

1: **Procedure:** Majority voting on dataset X , labels Y

2: **Split** X and Y into training and validation sets

3: **Bayesian Optimization:**

4: Optimise SVM hyperparameters using Bayesian search on the training set

5: Optimise DT hyperparameters using Bayesian search on training set

6: Optimise RF hyperparameters using Bayesian search on training set

7: **Train models with optimized hyperparameters:**

8: Train SVM model on training set using best parameters

9: Train DT model on training set using best parameters

10: Train RF model on training set using best parameters

11: Initialise ensemble predictions array E

12: **for** $i = 1$ to n **do** ▷ Loop through validation set

13: $svm_prediction \leftarrow SVM.predict(X[i])$

14: $dt_prediction \leftarrow DT.predict(X[i])$

15: $rf_prediction \leftarrow RF.predict(X[i])$

16: $ensemble_prediction \leftarrow MajorityVote$

17: $(svm_prediction, dt_prediction, rf_prediction)$

18: Append $ensemble_prediction$ to E

19: **end for**

20: Evaluate ensemble model performance on validation set

21: **Output:** Safe operating area

Algorithm 5. Pseudo code for proposed IFD model with Bayesian hyperparameter tuning.

Optimal values	
Parameter	Value
SVM	
C	10
kernel	rbf
gamma	0.01
Decision tree	
max_depth	10
min_samples_split	5
criterion	gini
Random forest	
n_estimators	150
max_depth	12
min_samples_leaf	2

Table 4. Hyperparameter tuning using Bayesian optimisation.

Scientific Reports | (2025) 15:32190

| <https://doi.org/10.1038/s41598-025-17145-4>

nature portfolio

10

Results and discussion

This research provides valuable insights into the EV industry and serves as a foundation for advancements in battery fault detection and BMS. The safe operating area of a Li-ion cell is determined by its temperature and voltage range. To facilitate battery grouping, K-means clustering is applied to categorise batteries based on similarities in their temperature and voltage characteristics. Ensemble methods are utilised to detect anomalies in battery performance. A battery displaying temperature and voltage deviations significantly different from others in the group may indicate specific conditions related to safe or unsafe operation. The implementation of this research requires minimal hardware, with a system specification of 4 GB RAM and an Intel i3 processor. The work is conducted using Python 3.9.0, leveraging libraries such as NumPy, pandas, torch, SNF, sklearn, and matplotlib. The scikit-learn package is primarily used for algorithm implementation.

Performance parameters

To evaluate this model's effectiveness, four performance criteria, which are accuracy, recall, precision, and F1-score, have been employed. The abbreviations TP, TN, FP, and FN in binary classification commonly represent True Positive, True Negative, False Positive, and False Negative, respectively. These parameters are quantified using the following mathematical expressions:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}, \quad (4)$$

$$Recall(R) = \frac{TP}{TP + FN}, \quad (5)$$

$$Precision(P) = \frac{TP}{TP + FP}, \quad (6)$$

$$F1 - score = \frac{2 \times P \times R}{P + R}. \quad (7)$$

Results

It is essential to comprehend the implementation prerequisites of an ensemble in real-world applications. An example of how ensemble approaches can be practically applied in fault classification is by combining many separate classifiers to enhance overall predicted accuracy and resilience. Once it comes to defect detection, where it is crucial to identify abnormal conditions accurately, ensemble approaches like RF, DT, and SVM are used. Ensembled approaches function by independently training numerous base classifiers and then combining their predictions to conclude. Within fault classification, individual base classifiers specialise in identifying specific fault patterns or features, enhancing the overall comprehension and nuanced perception of potential concerns. The collective decision-making of the ensemble is generally more dependable than that of individual classifiers, as it reduces the likelihood of overfitting specific training data outliers. It offers significant benefits when working with imbalanced datasets or complex fault scenarios since it effectively tackles these issues by utilising various models. Ensemble classifiers improve the overall dependability and precision of fault classification systems by working together, which makes them suitable for real-world applications where misclassification might have significant repercussions.

Figure 4 illustrates the normal working state of the battery, Fig. 4a shows the DS1 result, and Fig. 4b represents the result of DS2 for the temperature upper limit and lower limit. The battery is operated safely, and the unsafe area for under and over-temperature is marked as blue, which provides a safe environment to use the battery within the limit. For over-temperature, monitoring batteries is crucial to prevent overheating, which can accelerate degradation and compromise safety. Over-temperature protection ensures that the battery is not operated in conditions that exceed safe temperature thresholds, safeguarding both the battery and the surrounding environment. Whereas, under temperature accelerates in cold environments, batteries experience reduced performance or damage if operated below certain temperature thresholds. Under-temperature protection prevents the battery from being used in certain conditions.

Figure 5 illustrates the area of voltage safety for operation using two distinct actual data sets. Figure 5a displays the DS1 results, whereas Fig. 5b shows the DS2 results and the unsafe area for the battery is marked as yellow. In particular, this abrupt voltage breakdown may be attributed to a faulty connection, short circuit, collision, and other related causes. In the case of approaches that rely on a model threshold, it is possible that the abrupt decline in voltage may not have reached the prescribed limit, resulting in the possibility of detection failure. Under-voltage Protection setting a lower threshold for voltage helps prevent the battery from being discharged beyond a safe limit. Falling below this threshold can lead to diminished performance, irreversible damage, or safety hazards. Under-voltage protection ensures the battery is not over-discharged, preserving its overall health. Similarly, overvoltage protection, establishing an upper voltage limit, protects the battery from overcharging, which can result in overheating and electrolyte degradation and, in extreme cases, pose a safety risk. Over-voltage protection prevents the battery voltage from exceeding a level that could compromise its integrity. In the event of an atypical circumstance, it is imperative that the fault detection algorithm possesses the capability to accurately and expeditiously identify the defective battery.

The performance parameters of the algorithm using the ensemble are shown in Table 5. Battery fault detection falls within the acceptable range. While a fault is present, each fault indicator exhibits a distinct pattern.

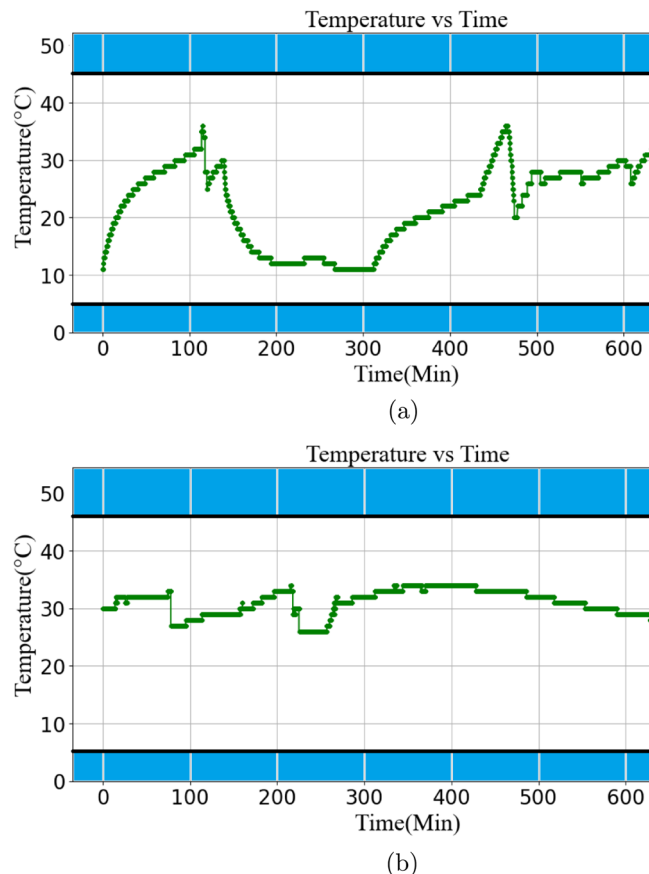


Fig. 4. (a) Temperature safe and unsafe using DS1. (b) Temperature vs time using DS2.

Testing of fault dataset on proposed model

For the evaluation of our proposed model, it is important to note that the battery dataset is currently operating without faults. In order to evaluate the algorithm, a faulty data set is generated that simulates charging and discharging scenarios. For example, during the charging mode, abrupt changes in voltage are induced by the power source, which in turn leads to fluctuations. The change in temperature is also generated by the increase in battery temperature due to a heat source. The data set contains balanced values with both negative and positive data samples, with 51.5 V as the median value for voltage, so above this value are positive samples and below this value are negative samples. Similarly, for temperature, the neutral temperature is 33.5 °C. When the operation value exceeds its threshold, it turns off the battery and restarts the operation.

F1, F2,..., and Fn depict the faults that occur for voltage and temperature. The voltage threshold we selected for protection is an upper limit of 55 V and a lower limit of 48V. Whereas the upper threshold for temperature is 45 °C and the lower threshold is 10 °C. Fig. 6a,b depict the upper and lower limits for voltage⁴⁰. This study used evaluation criteria to assess and compare the model's performance. Understanding the implementation requirements of the ensemble model in practical applications is crucial. Figure 7a,b represent the temperature upper limit and lower limit⁴¹.

The approach that necessitates matrix inversion and intensive calculations in IFD typically avoids these operations. Using appropriate training data and suitable kernel functions, the proposed predictive model efficiently calculates the IFD in a reduced time frame and with minimal storage requirements. When the inputs exceed the voltage level beyond the limiter value, the battery cuts off the load and regains its original position. A reset signal is generated and applied to the controller so that it can restart battery operation.

The dataset includes several borderline cases where fault indicators are either near or exactly at the detection threshold. The IFD model does not trigger a fault detection when conditions are at or just below the threshold. However, once the parameters exceed this threshold, even slightly, the model reliably identifies the fault. This behaviour highlights a deliberate balance between avoiding false positives and maintaining accurate fault detection.

Comparative analysis

The detection of voltage issues in a lithium-ion battery pack by employing an outlier detection technique. The battery dynamics are represented through the utilisation of the ECM, while the parameters are determined by employing the EKF technique²⁴. Consequently, by establishing a correlation between the states of the battery and the characteristics of the model, the objective of detecting problems inside the battery pack is effectively

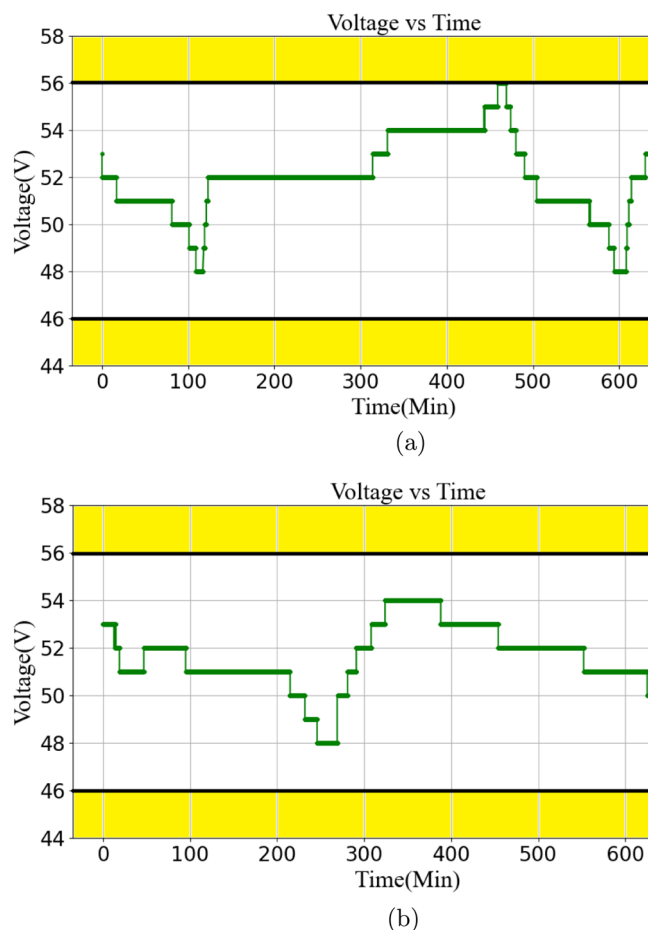


Fig. 5. (a) Voltage safe and unsafe using DS1. (b) Voltage vs time using DS2.

Model	Accuracy	Precision	Recall	F1-score
SVM	0.72	0.69	0.74	0.71
DT	0.82	0.83	0.83	0.80
RF	0.89	0.91	0.91	0.90
Ensemble (IFD)	0.94	0.93	0.95	0.94

Table 5. Performance evaluation of proposed model.

converted into the objective of recognising abnormalities within the set of parameters. To tackle this matter, a methodology utilising a distance-based approach is implemented for outlier detection.

The defect is remedied by evaluating whether the fault indicator exceeds the predetermined threshold. In order to evaluate the effectiveness of the proposed technique, simulations and test bench experiments are conducted. The SVMs demonstrated efficiency in analysing real-world datasets, exhibiting superior performance with an accuracy rate²⁵. The SVM has emerged as a prominent technique in pattern recognition, garnering substantial attention and engagement from researchers. The selection of this specific method has been determined to produce superior results compared to other classifications in this particular application. This study used evaluation criteria to assess and compare the model's performance. The accuracy of the model is observed at 0.91. Figure 8 compares the model with existing work.

Finally, the ensemble fault detection model provides much better output, which is about 94% and provides a more robust and accurate prediction on a variety of data patterns. The ensemble's success often relies on the diversity of the constituent models and the effective combination of their predictions.

Limitations of the proposed work

Despite the significant progress, some flaws in the research should be considered. For the ensemble model algorithm to perform well, it requires high-quality data, since it is already robust and accurate. The accuracy of predictions can be minimally affected if datasets in the real world are very sparse. It could also be optimised

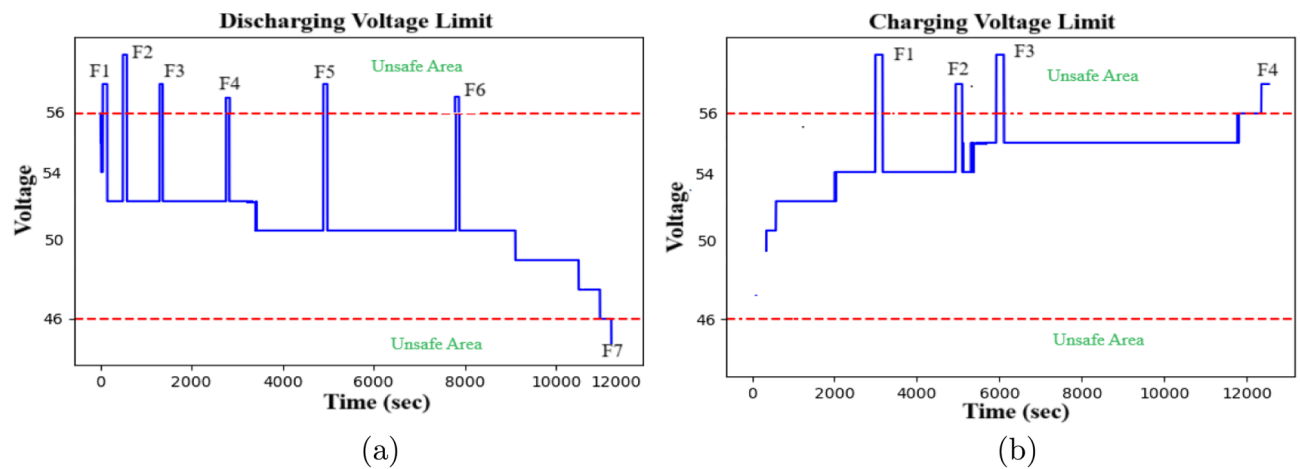


Fig. 6. Safe and unsafe area—(a) Voltage limiter (discharging). (b) Voltage limiter (charging).

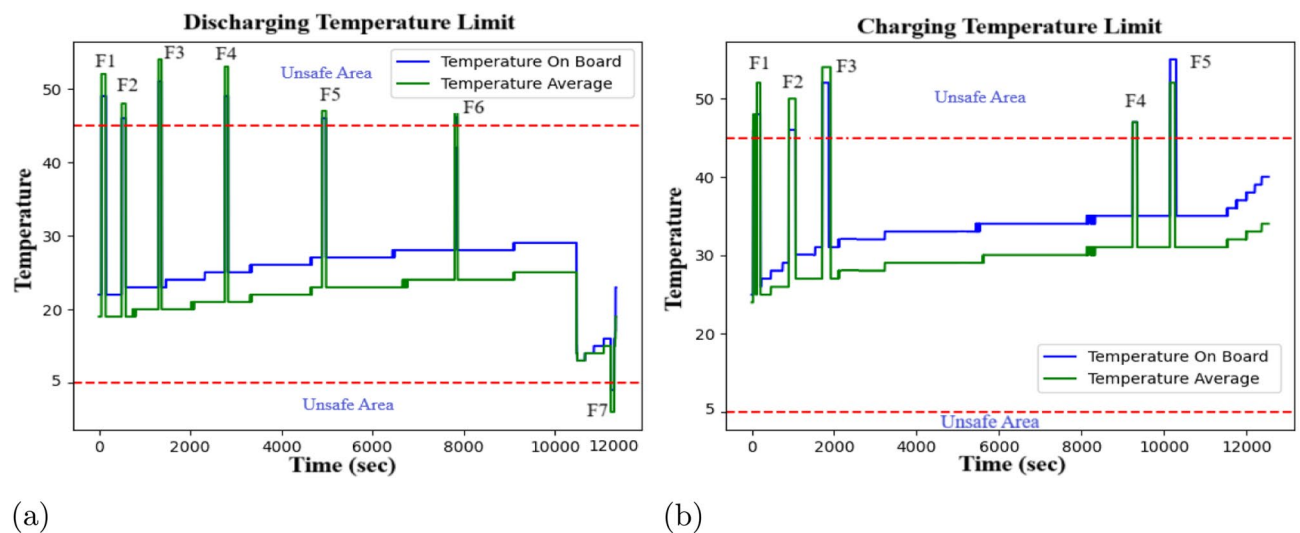


Fig. 7. Safe and unsafe area—(a) Temperature for discharging. (b) Temperature for charging.

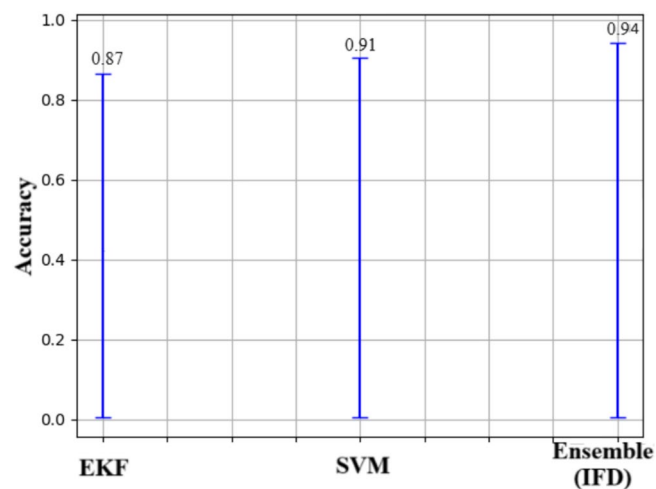


Fig. 8. Comparison of proposed model with existing work.

to work better and more efficiently with larger applications or systems that have limited resources. Changes in calibration and several features could help apply the framework to lithium-ion batteries, for which it was primarily developed. All in all, it is crucial to carry out further research to prove that this framework can easily work in situations with tight and varying schedules to ensure prompt adjustment.

Conclusion

In this paper, we emphasised the critical importance of accurate fault estimation in batteries, especially for electric and hybrid electric vehicles, where safety and performance are paramount. The proposed IFD framework enhances safety by effectively identifying outliers and linearly separable faulty patterns within the dataset. Our approach leverages real-time system data, such as voltage and temperature collected during controlled charging and discharging cycles of a 51.8 V, 100 Ah lithium-ion battery. Key techniques employed include K-means clustering to identify behaviour classes and an ensemble learning strategy to classify safe versus unsafe operating conditions. The fault detection model, trained with carefully selected data and kernel functions, demonstrates strong predictive capability in identifying potential faults. By integrating machine learning models, the framework not only improves fault detection accuracy but also establishes meaningful correlations between safe operational behaviour and battery ageing. Ultimately, the IFD model serves as a valuable tool for determining optimal battery usage conditions, contributing to enhanced safety, extended battery life, and more reliable electric vehicle performance.

Data availability

The dataset generated and analysed in this study is available from the corresponding author upon request (Rahul Kumar Kamboj).

Received: 29 May 2025; Accepted: 21 August 2025

Published online: 01 September 2025

References

- Cheng, K. W. E. *Recent Development on Electric Vehicles* 1–5 (2009)
- Alanazi, F. Electric vehicles: Benefits, challenges, and potential solutions for widespread adaptation. *Appl. Sci.* **13**(10), 16. <https://doi.org/10.3390/app13106016> (2023).
- Liu, W., Placke, T. & Chau, K. T. Overview of batteries and battery management for electric vehicles. *Energy Rep.* **8**, 4058–4084. <https://doi.org/10.1016/j.egy.2022.03.016> (2022).
- Tran, M.-K. & Fowler, M. A review of lithium-ion battery fault diagnostic algorithms: Current progress and future challenges. *Algorithms* **13**(3), 62. <https://doi.org/10.3390/a13030062> (2020).
- Xie, K. et al. A method for measuring and evaluating the fault response performance of battery management system. *Energy Rep.* **8**, 639–649. <https://doi.org/10.1016/j.egy.2022.01.223> (2022).
- Hong, J., Zhang, H. & Xu, X. Thermal fault prognosis of lithium-ion batteries in real-world electric vehicles using self-attention mechanism networks. *Appl. Therm. Eng.* **226**, 120304. <https://doi.org/10.1016/j.applthermaleng.2023.120304> (2023).
- Abada, S. et al. Safety focused modeling of lithium-ion batteries: A review. *J. Power Sources* **306**, 178–192. <https://doi.org/10.1016/j.jpowsour.2015.11.100> (2016).
- Jiang, L., Diao, X., Zhang, Y., Zhang, J. & Li, T. Review of the charging safety and charging safety protection of electric vehicles. *World Electr. Veh. J.* **12**(4), 184. <https://doi.org/10.3390/wevj12040184> (2021).
- Meissner, E. & Richter, G. Battery monitoring and electrical energy management: Precondition for future vehicle electric power systems. *J. Power Sources* **116**(1), 79–98. [https://doi.org/10.1016/S0378-7753\(02\)00713-9](https://doi.org/10.1016/S0378-7753(02)00713-9) (2003).
- Chen, C. H. et al. Development of experimental techniques for parameterization of multi-scale lithium-ion battery models. *J. Electrochem. Soc.* **167**(8), 080534. <https://doi.org/10.1149/1945-7111/ab9050> (2020).
- Wang, Z., Hong, J., Liu, P. & Zhang, L. Voltage fault diagnosis and prognosis of battery systems based on entropy and z-score for electric vehicles. *Appl. Energy* **196**, 289–302. <https://doi.org/10.1016/j.apenergy.2016.12.143> (2017).
- Zhou, Z. et al. Experimental analysis of lengthwise/transversal thermal characteristics and jet flow of large-format prismatic lithium-ion battery. *Appl. Therm. Eng.* **195**, 117244. <https://doi.org/10.1016/j.applthermaleng.2021.117244> (2021).
- Chen, Z., Xu, K., Wei, J. & Dong, G. Voltage fault detection for lithium-ion battery pack using local outlier factor. *Measurement* **146**, 544–556. <https://doi.org/10.1016/j.measurement.2019.06.052> (2019).
- Zhang, L., Mu, Z. & Sun, C. Remaining useful life prediction for lithium-ion batteries based on exponential model and particle filter. *IEEE Access* **6**, 17729–17740. <https://doi.org/10.1109/ACCESS.2018.2816684> (2018).
- Wei, J., Dong, G. & Chen, Z. Remaining useful life prediction and state of health diagnosis for lithium-ion batteries using particle filter and support vector regression. *IEEE Trans. Ind. Electron.* **65**(7), 5634–5643. <https://doi.org/10.1109/TIE.2017.2782224> (2018).
- Zhao, R., Liu, J. & Gu, J. Simulation and experimental study on lithium ion battery short circuit. *Appl. Energy* **173**, 29–39. <https://doi.org/10.1016/j.apenergy.2016.04.016> (2016).
- Feng, X. et al. Thermal runaway mechanism of lithium ion battery for electric vehicles: A review. *Energy Storage Mater.* **10**, 246–267. <https://doi.org/10.1016/j.ensm.2017.05.013> (2018).
- Mohsin Abdulazeez, A. *Machine Learning Applications Based on Svm Classification for Libs Batteries: A Review* (2021)
- Zhang, X., Li, Z., Luo, L., Fan, Y. & Du, Z. A review on thermal management of lithium-ion batteries for electric vehicles. *Energy* **238**, 121652. <https://doi.org/10.1016/j.energy.2021.121652> (2022).
- Piao, C., Huang, Z., Su, L. & Lu, S. Research on outlier detection algorithm for evaluation of battery system safety. *Adv. Mech. Eng.* **6**, 830402. <https://doi.org/10.1155/2014/830402> (2014).
- Ma, H., Hu, Y. & Shi, H. Fault detection and identification based on the neighborhood standardized local outlier factor method. *Ind. Eng. Chem. Res.* **52**(6), 2389–2402. <https://doi.org/10.1021/ie302042c> (2013).
- Ferdowsi, H., Jagannathan, S. & Zawodniok, M. An online outlier identification and removal scheme for improving fault detection performance. *IEEE Trans. Neural Netw. Learn. Syst.* **25**(5), 908–919. <https://doi.org/10.1109/TNNLS.2013.2283456> (2013).
- Hu, L. Y., Huang, M.-W., Ke, S.-W. & Tsai, C.-F. The distance function effect on k-nearest neighbor classification for medical datasets. *SpringerPlus* **5**, 7. <https://doi.org/10.1186/s40064-016-2941-7> (2016).
- Yang, R., Xiong, R., Shen, W. & Lin, X. Extreme learning machine-based thermal model for lithium-ion batteries of electric vehicles under external short circuit. *Engineering* **7**(3), 395–405. <https://doi.org/10.1016/j.eng.2020.08.015> (2021).
- Lskafi, M. A. *Fault Diagnosis and Failure Prognostics of Lithium-Ion Battery Based on Least Squares Support Vector Machine and Memory Particle Filter Framework*.

26. Li, D., Zhang, Z., Liu, P., Wang, Z. & Zhang, L. Battery fault diagnosis for electric vehicles based on voltage abnormality by combining the long short-term memory neural network and the equivalent circuit model. *IEEE Trans. Power Electron.* **36**(2), 1303–1315. <https://doi.org/10.1109/TPEL.2020.3008194> (2021).
27. Yüksel, G. & Alkaya, A. A novel state of health estimation approach based on polynomial model for lithium-ion batteries. *Int. J. Electrochem. Sci.* **18**(5), 100111. <https://doi.org/10.1016/j.ijoes.2023.100111> (2023).
28. Lale, T. A novel energy management technique in electric vehicles through temperature-driven c-rate. *Electr. Power Syst. Res.* **247**, 111748. <https://doi.org/10.1016/j.epsr.2025.111748> (2025).
29. Yamaçlı, V. State-of-health estimation and classification of series-connected batteries by using deep learning based hybrid decision approach. *Heliyon* **10**(20), 39121. <https://doi.org/10.1016/j.heliyon.2024.e39121> (2024).
30. Shang, Y., Wang, S., Tang, N., Fu, Y. & Wang, K. Research progress in fault detection of battery systems: A review. *J. Energy Storage* **98**, 113079. <https://doi.org/10.1016/j.est.2024.113079> (2024).
31. Hong, S., Kang, M., Park, H., Kim, J. & Baek, J. Real-time state-of-charge estimation using an embedded board for li-ion batteries. *Electronics* **11**(13), 10. <https://doi.org/10.3390/electronics11132010> (2022).
32. Obaid, H. S., Dheyab, S. A. & Sabry, S. S. The impact of data pre-processing techniques and dimensionality reduction on the accuracy of machine learning. In *2019 9th Annual Information Technology, Electromechanical Engineering and Microelectronics Conference (IEMECON)* 279–283. <https://doi.org/10.1109/IEMECONX.2019.8877011> (2019).
33. Na, S., Xumin, L. & Yong, G. Research on k-means clustering algorithm: An improved k-means clustering algorithm. In *2010 Third International Symposium on Intelligent Information Technology and Security Informatics* 63–67. <https://doi.org/10.1109/IITSI.2010.74> (2010).
34. Li, Y. & Wu, H. A clustering method based on k-means algorithm. *Phys. Procedia* **25**, 1104–1109. <https://doi.org/10.1016/j.phpro.2012.03.206> (2012).
35. Singh, A. et al. ediapredict: An ensemble-based framework for diabetes prediction. *ACM Trans. Multimed. Comput. Commun. Appl.* **17**, 1–26. <https://doi.org/10.1145/3415155> (2021).
36. Cervantes, J. et al. A comprehensive survey on support vector machine classification: Applications, challenges and trends. *Neurocomputing* **408**, 118. <https://doi.org/10.1016/j.neucom.2019.10.118> (2020).
37. Samantaray, S., Kamwa, I. & Joos, G. Decision tree based fault detection and classification in distance relaying. *Int. J. Eng. Intell. Syst. Electric. Eng. Commun.* **19**, 139 (2011).
38. Sarica, A., Cerasa, A. & Quattrone, A. Random forest algorithm for the classification of neuroimaging data in Alzheimer's disease: A systematic review. *Front. Aging Neurosci.* **9**, 329. <https://doi.org/10.3389/fnagi.2017.00329> (2017).
39. Jaiswal, J. K. & Samikannu, R. Application of random forest algorithm on feature subset selection and classification and regression. In *2017 World Congress on Computing and Communication Technologies (WCCCT)* 65–68. <https://doi.org/10.1109/WCCCT.2016.25> (2017).
40. Liu, X. et al. Thermal runaway of lithium-ion batteries without internal short circuit. *Joule* **2**, 15. <https://doi.org/10.1016/j.joule.2018.06.015> (2018).
41. Zhang, Z., Liang, L. & Fei, H. Investigation on safe-operating-area degradation and failure modes of sic mosfets under repetitive short-circuit conditions. *Power Electron. Devices Compon.* **4**, 100026. <https://doi.org/10.1016/j.pedc.2022.100026> (2023).

Author contributions

Rahul Kumar kamboj and Ashima Singh designed the study. Anju Bala collected the data. Rahul Kumar Kamboj and Anju Bala analysed the data. Mukesh Singh, Ashima Singh and Rahul collaborated on manuscript writing. Mukesh Singh obtained funding for the study. All authors read and approved the final manuscript.

Funding

The authors gratefully acknowledge the Ministry of Electronics and Information Technology (MeitY), Government of India, for the support through Project Identification No. 25(2)/2021-ESDA.

Declarations

Competing interests

The authors declare no competing interests.

Additional information

Correspondence and requests for materials should be addressed to R.K.K.

Reprints and permissions information is available at www.nature.com/reprints.

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International License, which permits any non-commercial use, sharing, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if you modified the licensed material. You do not have permission under this licence to share adapted material derived from this article or parts of it. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

© The Author(s) 2025