

An optimized ensemble framework for machinery fault detection in IoT environments

Received: 13 September 2025

Accepted: 12 February 2026

Published online: 24 February 2026

Cite this article as: Devi Gayadri S.V., Kanagaraj G., Giri J. *et al.* An optimized ensemble framework for machinery fault detection in IoT environments. *Sci Rep* (2026). <https://doi.org/10.1038/s41598-026-40335-7>

S. V. Devi Gayadri, G. Kanagaraj, Jayant Giri & Mohammad Kanan

We are providing an unedited version of this manuscript to give early access to its findings. Before final publication, the manuscript will undergo further editing. Please note there may be errors present which affect the content, and all legal disclaimers apply.

If this paper is publishing under a Transparent Peer Review model then Peer Review reports will publish with the final article.

ARTICLE IN PRESS

An Optimized Ensemble Framework for Machinery Fault Detection in IoT Environments

Devi Gayadri S V¹, Kanagaraj G^{*2}, Jayant Giri^{3,4,5} Mohammad Kanan ^{*6,7}

¹Research Scholar, Department of Mechatronics Engineering, Thiagarajar College of Engineering, Madurai, Tamil Nadu 625015, India, rithyakumar@gmail.com

^{*2}Professor, Department of Mechatronics Engineering, Thiagarajar College of Engineering, Madurai, Tamil Nadu 625015, India, gkmech@tce.edu

³Department of Mechanical Engineering, Yeshwantrao Chavan College of Engineering, Nagpur, India, jayantpgiri@gmail.com

⁴Division of Research and Development, Lovely Professional University, Phagwara, India

⁶ Centre for Research Impact & Outcome, Chitkara University Institute of Engineering and Technology, Chitkara University, Rajpura, 140401, Punjab, India

^{*6}Department of Industrial Engineering, College of Engineering, University of Business and Technology, Jeddah 21448, Saudi Arabia, m.kanan@ubt.edu.sa

⁷Department of Mechanical Engineering, College of Engineering, Zarqa University, Zarqa, Jordan

Abstract:

Fault detection in IoT-enabled machinery involves identifying defects in the operation of industrial equipment to foil breakdowns that ensure reliability. It typically relies on analysing data from IoT sensors that monitor key parameters such as temperature, vibration, pressure, and speed in machinery. Fault detection systems in industrial settings often face challenges due to the inconsistent distribution of critical sensor data, affecting monitoring systems reliability. Thus, this research aims to design an Optimized Robust PCA-based Ensemble framework for Machine Learning model to find the abnormalities in the IoT-enabled machinery and ensure robust performance by analysing the distribution patterns of critical sensors. This framework is used to extract the essential samples based on voltage, speed, temperature and vibrations. The ensemble learning model includes K-Nearest Neighbour (KNN) and Adaboost, which are optimized by Bayesian optimization. The importance of optimization technique used to fine-tune the parameters of ensemble learning model. The improved ensembled learning model detects the fault in the IoT enabled machinery can be ensured by the performance metrics such as accuracy, precision, and detection rate and false negative rate. The effectiveness of the proposed model is evaluated by varying operational conditions and viability is analysed using an extensive dataset is IoT enabled machinery operations. Thus, the experimentation of the proposed model significantly improves fault detection reliability, ensuring operational efficiency and reducing downtime in industrial infrastructures.

Keywords: Fault Detection, Machine Learning models, Industrial Machinery, Sensors, Reliability, Enhanced Feature extraction and optimization

1. Introduction:

Machinery is the backbone of industrial control systems, driving efficiency, precision, and reliability across various operations. In the era of IoT-based environments, industrial machinery has evolved to integrate smart sensor networks, enabling real-time monitoring and enhanced decision-making. These advancements have significantly improved the ability of machines to perform complex tasks with greater accuracy and consistency. However, the optimal performance of IoT-enabled systems depends on the proper functioning of critical factors such as operating temperatures, sensor accuracy, and vibration control [1 & 2]. IoT sensors play a pivotal role in maintaining these parameters by continuously monitoring operating conditions and transmitting data for

analysis. For instance, maintaining optimal temperatures is crucial to prevent overheating, which can lead to severe damage or reduced equipment lifespan. Real-time data from temperature sensors allows for proactive intervention to avoid potential failures. Similarly, IoT-based vibration sensors can detect anomalies caused by misaligned or deteriorating components, ensuring timely maintenance and preserving precision in operations. Faults in these critical areas can lead to cascading effects such as unplanned downtime, increased maintenance costs, and reduced productivity. In IoT-enabled industrial systems, addressing these faults with advanced fault detection and predictive maintenance technologies is essential for ensuring operational efficiency, minimizing disruptions, and maintaining overall profitability [4, 43-44].

Despite significant advancements in fault detection, existing studies in industrial systems have revealed several limitations. Many studies have focused on specific fault types, such as bearing failures or vibration analysis, often neglecting a comprehensive approach that integrates multiple fault parameters. Additionally, traditional methods, like time-domain and frequency-domain analysis, struggle with accurately diagnosing machinery faults in complex and dynamic industrial infrastructure [3-5]. Another limitation is the reliance on handcrafted feature extraction techniques, which require extensive domain knowledge and may not capture subtle patterns indicative of impending failures. Furthermore, the impact of external factors like varying operational conditions, sensor noise, and environmental influences is often underexplored, leading to potential inaccuracies in fault detection [6 & 7]. As a result, there is a growing need for an advanced model tailored for fault analysis. In recent years, intelligent detection systems have become crucial across various fields, with Machine Learning models demonstrating exceptional performance in this domain.

Machine Learning models are effective in the aspect of classification and detection. Enormous research showcased the usage of those models in identifying faults in industrial machinery. Hence, this research article aims to develop a machine learning-based fault detection system that ensures effective detection for faults in machinery in industrial infrastructure [8-10]. Models such as Support Vector Machines (SVM), Decision Trees (DT), Random Forests (RF), and k-nearest Neighbours (KNN) have been extensively explored for fault classification and anomaly detection. These models are known for their simplicity, interpretability, and robustness in detecting faults across various operational conditions. Hence, various machine learning models have been proposed for fault detection in machinery components [11, 12]. However, each models pose difficulties in multi class classification. Therefore, to overcome the issues, an optimized robust PCA-based ensemble learning model is proposed. The major contribution of the model is highlighted below,

- An optimized robust PCA technique is utilized to extract high-quality features from noisy and high-dimensional industrial machinery data
- Stage-wise feature selectors improve multiclass detection by identifying the most pertinent features at each phase of the detection process, thereby improving the accuracy and efficiency of sensor devices
- Ensemble model (KNN + Adaboost) provides adaptability and resilient predictive capabilities for diagnosing faults in IoT-enabled settings
- The integration of KNN for localized decision-making with the adaptive features of AdaBoost provides an effective strategy for fault detection
- Bayesian optimisation effectively navigates the hyperparameter space, thereby minimising the number of iterations and computational resources required

The structure of the article is as follows: Section 2 deliberates the recent research in advancements towards fault detection. Section 3 provides a detailed explanation of the proposed model. Section 4 delineates the experiment details. Section 5 discusses the

experimental results and their implications, while Section 6 concludes the paper and outlines potential directions for future work.

2. Related Works:

This section briefly discusses the recent studies on fault detection in industrial machinery that demonstrate significant advancements through Machine Learning (ML) techniques, focusing on enhancing operational reliability and efficiency. Various approaches leverage ensemble models, deep learning, and IoT-based systems to analyse sensor data and detect faults in the machineries. Owing to this, some pertinent works are discussed below, along with their limitations and scope for the proposed model.

Dhiraj Neupane et al. integrate machine learning approaches [11] for early equipment fault detection. It emphasizes how modern sensors and big data can improve diagnostic capabilities. However, its inability to manage sensor noise and adjust to various operating circumstances limits its dependability in dynamic situations. Moshrefi et al. [12] use ensemble machine learning models (gradient boosting and stacking) with ultrasonic signal processing for flaw detection. Though accuracy is increased using ensemble models, the research does not include real-world deployment scenarios, and computational efficiency is unaddressed. Basangar et al. [13] tackle defects in rotating industrial gear with a deep learning focus. Despite increasing accuracy, low-resource resources are considered industrial contexts and are limited by their high computing cost and need for huge datasets. Wei Cui et al. [14] blend methods such as Deep Belief Networks, Autoencoders, Convolutional Neural Networks, Recurrent Neural Networks, Transfer learning, and Generative Adversarial Networks. Though the model's scalability is good, the proposed model failed to examine the effects of changing operating circumstances and inconsistent sensors. Laaradj et al. [15] use ML to classify faults in spinning machinery using vibration signals. Although the approach shows better results, its limited feature extraction capabilities make it difficult to uncover small flaws. The significance of feature engineering and selection in enhancing ML model performance for fault detection is examined in this work. However, manual feature extraction makes model deployment more complex and calls for domain expertise.

Sohaib et al. [16] Apply advanced machine learning and image processing techniques for gearbox fault detection, achieving high accuracy (96%) with innovative methods like edge detection and maximized pooling. The study is based on vibration signal images, offering insights into using visual data for diagnostics. However, it is limited to predefined datasets and specific machinery types, reducing generalizability to other industrial applications. Qiu et al. [17] Reviews the use of deep learning models, including emerging architectures like transformers and graph neural networks, for fault diagnosis and prognosis in industrial systems. While it highlights the advancements in feature representation and fault detection, challenges like data imbalance, multimodal data integration, and edge device implementation remain unaddressed. Moosavi et al. [18] integrate explainable artificial intelligence (XAI) approaches with traditional machine learning for fault detection, improving interpretability and trust in model predictions. Despite the promising results, these models' computational complexity and dependence on huge datasets are identified as challenges. Perumal et.al. [19] Explored the application of supervised machine learning models for detecting faults in industrial machinery. It highlights techniques for processing IoT sensor data, optimizing detection performance, and addressing challenges related to data imbalance and noise interference. Priya et.al. Analysed fault detection in wireless sensor networks leveraging machine learning algorithms to enhance fault tolerance and system reliability [20]. Leite et.al. Introduced an AutoML-based real-time fault detection and diagnosis system optimized for IoT environments. The [21] system leverages discrete-event and continuous-variable features for comprehensive fault detection. Ellah et.al. Explained machine learning methods for fault detection in wireless sensor networks, which are fundamental to IoT environments. It focused on techniques that prove the reliability and accuracy of fault detection

mechanisms, addressing constraints like limited computational resources and varying environmental conditions [22]. Inyang et.al. [23] proposed a deep blended ensemble learning model that integrated the optimized signal transformers. A compound dataset is used for training the model to diagnose the faults in rotating machines like gearboxes, bearings and shafts. Miao et.al. [24] focused on a feature-driven ensemble learning method that integrated time and frequency domain information for fault detection in industrial processes. Bayesian inference combines results from feature extraction layers, achieving superior performance on the Tennessee Eastman Process dataset. Jose et.al. [25] proposed ensemble learning approach for bearing fault detection. Various algorithms are considered for observation among KNN, XGBoost and SVM models are considered and ensemble for fault detection. Siddique et.al [26] proposed a Burst Aware Adaptive Frame Segmentation based Weighted Stacked ensemble model for Acoustic emission and the model achieved 99.21% of accuracy. However, the authors failed to mention the overfitting issues, computational complexity issues and also the dataset reliability is not verified. Siddique et.al [27] proposed fault detection model with transformers based semi-supervised model and achieved an accuracy rate of 99.68%. The model was tested using the benchmark dataset and the real time valuation is not discussed which lead to the limitations of data reliability. Zaman et.al [28] proposed a hybrid framework for fault detection mechanism using logarithmic continuous wavelet scalograms, Canny edge operator, and dual branch encoder. Despite this framework, it has limitations such as dataset reliability, limited generalizability, overfitting issues and complexity in tuning the parameters of the proposed models. Zahoor et.al [29] proposed a framework for fault diagnosis of centrifugal pumps using various methods however failed to mention the uncertainty of other fault types and pumps.

The reviewed works highlighted the significant advancements in fault detection using machine learning techniques for industrial machinery. However, several limitations persist across these studies (i) some models failed to address sensor noise and dynamic operating conditions, reducing their approach's adaptability and reliability. (ii) Enhanced accuracy with ensemble models, overlook computational efficiency and real-world deployment feasibility. Overall, these limitations underline the need for models that are computationally efficient, generalizable, and robust to diverse conditions.

3. Proposed Methodology

The proposed framework of Optimized RPC and Ensemble learning model shown in Fig.1 is used to detect faults in industrial machinery and prevent further defects. Though various research scopes focus on ensemble-based solutions, most of the works failed to address the problem of computational efficiency, pre-processing model, data dependencies, and scalability. To address all these issues, an **optimized Robust Principal Component Analysis based Ensemble Machine Learning model (R-PCA based EML)** is proposed and validated using real-time data collected from the simulated environment for a period. The working of the proposed method is categorized into three subsections: (i) Data Pre-processing, (ii) Feature selection & Feature extraction, and (iii) Optimized Ensemble Machine Learning based fault detection model and discussed in the following subsections.

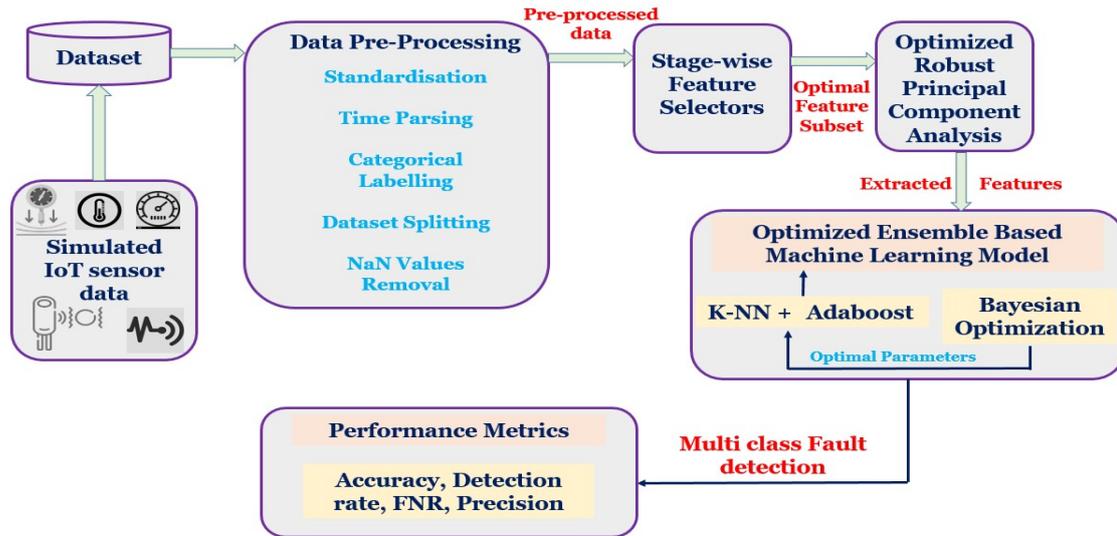


Fig.1 Proposed Framework of Optimized RPCA and Ensemble Learning model

3.1. Data Collection:

The dataset is collected using a real-time IoT monitoring simulated environment where wireless sensors are strategically deployed to continuously monitor critical machine parameters, including Temperature, Pressure, Speed, Power Supply Voltage, and Vibration. These parameters provide real-time insights into the machine's operational state, serving as a foundation for fault detection and health analysis. Each wireless sensor captures relevant data and transmits it wirelessly to the Wireless Sensor Network (WSN) module, which acts as a central hub. The WSN module relays the aggregated sensor data to a MyRIO-embedded device, which interfaces with a personal computer for further processing and analysis. Wireless communication ensures seamless monitoring, even in hazardous or remote environments, making the system highly adaptable to various industrial applications. The data captured by the sensors is timestamped and recorded using NI LabVIEW programming software. This platform provides a robust, flexible environment for real-time data acquisition, analysis, and visualization. The timestamped data is stored alongside the current date and time, facilitating historical analysis and trend identification. This is critical for predicting potential machine failures. By maintaining a comprehensive log of machine parameters, the system enables engineers to: (i) Analyse the machine's long-term performance and (ii) Detect gradual trends that may indicate future malfunctions. This systematic approach to data collection ensures a reliable and scalable framework for proactive machine monitoring and fault detection. The major motivation for the simulation of this dataset relies on systematically controlling the fault types, severity levels in the machinery systems, and operation managing sensors. This dataset ensures the consistency and generate the sufficient fault instances replicating the architect and statistical alignment of real time industrial environment.

The sensors are interfaced with the WiFi Module Kit through a breadboard and connecting wires. The WiFi Module Kit [Fig.2 (a) and (b)] is initially connected to a PC system for program development, including writing, compiling, and uploading. Once the program is successfully uploaded to the module, it is disconnected from the PC and prepared for data acquisition.



Fig.2 (a) CC3200 WiFi Launch, (b) Motion Drive Kit and PLC Based IOT Kit

3.2. Data Pre-processing:

Data pre-processing is imperative for training the model to handle sensor-based data effectively. These data contain null values, outliers, and inconsistencies, which degrade the learning model's performance. Hence, to ensure the reliable analysis of data, pre-processing is crucial. It includes data cleaning, data integration, data transformation, and so on. In our proposed model, we have included various data pre-processing techniques, which are described as follows,

(a) Standardisation

The data has been rescaled with the mean value of 0 and the standard deviation of 1 [28]. This is also called z-score scaling. The process is carried out using the formula given in formula (1).

$$sf' = (of - \text{mean}(of)) / (\text{sd}(of)) \quad (1)$$

Where sf' denotes standardised feature, of original feature and $\text{sd}(of)$ refers the standard deviation of original feature.

(b) Time Parsing

As most IoT sensor data consists of "date" and "time" as columns, this is converted to a single timestamp value or date-time format for consistency and easier indexing.

(c) Categorical labelling

Considered simulated IoT sensor data has features "vibration", "speed", "voltage," and "temperature." And their relative values. There are no specific labels for fault analysis. As per the domain and expertise knowledge, some criteria are framed for fault analysis (A detailed explanation of these criteria is given in the data description section). The labels are as follows: "**Very Low (VL)**", "**Low (L)**", "**Medium (M)**", "**High**" and "**Very High**".

(d) Dataset Splitting

Data splitting is an important process in the machine learning process as it helps in ensuring the model to avoid overfitting and under fitting [26]. Hence, the data is splitted as **training (80%) and testing data (20%)**. The model is trained using the training data and the testing data is unseen data. The model is robust when it performs well with testing data.

(e) NaN values removal

NaN values, or "Not a Number," play a crucial role in data analysis as they indicate missing or incomplete information in a dataset [27]. Effectively managing these values is essential to maintain the accuracy and reliability of analytical results. We used "*isnan()*" function from the python library to manage the NaN values in the considered dataset.

3.3. Feature selection

Feature selection is a crucial step in pre-processing, particularly for datasets containing multiple correlated attributes [29,30]. In this article, we propose a **Stage-Wise Feature Selection** methodology to identify optimal subsets of features that contribute significantly to fault detection in machinery. By adopting this method, we aim to enhance model performance and interpretability while reducing computational overhead. The feature selection process is structured into distinct stages, each evaluating a specific subset of the dataset. The subsets are constructed based on the combination of the timestamp feature, which provides temporal context, with other key attributes that represent critical operational parameters of the machinery. These stages are described in Table 1.

Table 1 Four Different Stages of IoT enabled machinery dataset

Stages			
Stage 1 (Voltage Vs Timestamp)	Stage 2 (Speed Vs Timestamp)	Stage 3 (Temperature Vs Timestamp)	Stage 4 (Vibration Vs Timestamp)
This subset focuses on the relationship between temporal changes and voltage variations. Voltage serves as a critical indicator of electrical stability and faults, making it vital for detecting anomalies in electrical systems.	This stage evaluates the interplay between time and speed, representing dynamic operational states of the machinery. Variations in speed over time can signify mechanical imbalances or performance degradation.	The third subset examines the correlation between time and temperature. Temperature variations often reflect thermal stress or overheating, which are precursors to machinery failure.	The final stage considers temporal data alongside vibration measurements. Vibration is a key diagnostic parameter for identifying mechanical wear and tear or misalignments.

By segmenting the dataset into these targeted feature subsets, the stage-wise selection method allows for a comprehensive analysis of how each parameter contributes to fault detection. This iterative approach ensures that the most relevant features for fault diagnosis are retained, while redundant or less significant features are excluded, thereby improving the model's diagnostic capabilities.

3.4. Feature extraction - Optimized Robust Principal Component Analysis

After feature selection, feature extraction becomes a pivotal step in preparing the dataset for fault detection tasks. Feature extraction reduces the dimensionality of the dataset while preserving the most informative components, ensuring the model's efficiency and robustness. Hence, an optimization-based Robust Principal Component Analysis (R-PCA) method [31-33 &35] is proposed to extract critical features from the selected subsets of the dataset.

RPCA decomposes the data matrix X into two components: one as low rank matrix LM which represents the underlying patterns of the data. Whereas the sparse matrix is represented as SM which captures the anomalies. This decomposition process is performed by solving the optimization problem in formula (2)

$$\min_{LM, SM} ||LM||_* + \lambda ||SM||_1 \text{ subject to } x = LM + SM \quad (2)$$

Where $||LM||_*$ denotes the nuclear form of LM . $||SM||_1$ represents the l_1 normalization of SM . The parameter λ controls the trade-off between the low-rank approximation and the sparsity of the noise component. The following subsection will explain how the R-PCA works with the considered machinery dataset.

3.5. Optimized Ensemble Machine Learning based fault detection model

Industrial systems rely heavily on robust fault detection mechanisms to ensure machinery operates efficiently without unexpected breakdowns. Fault detection involves identifying abnormalities that lead to system failures and to ensure timely intervention. With advancements in machine learning, ensemble models have become a preferred choice for tackling complex fault detection problems. Hence, we proposed an optimized ensemble machine learning model combining K-Nearest Neighbours (KNN) and AdaBoost classifiers. The model's performance is further enhanced through Bayesian Optimization, which fine-tunes hyper parameters of ensemble models, leading to improved multiclass fault detection.

KNN is a simple, non-parametric machine learning algorithm widely used for classification problems, including fault detection. The primary concept behind KNN is that similar data points exist close to each other in the feature space [34]. The algorithm assigns the label of the majority class among the nearest neighbours to a data point under consideration. AdaBoost (Adaptive Boosting) is an ensemble learning algorithm that combines multiple weak classifiers to create a strong classifier. In the context of fault detection, AdaBoost iteratively adjusts weights for data points, emphasizing those that were misclassified in previous iterations [36 -38].

The proposed model combines KNN and AdaBoost predictions through a weighted voting mechanism. KNN Captures localized patterns and provides probabilistic outputs for each class based on the nearest neighbours. While, AdaBoost Contribution focuses on hard-to-classify instances and assigns higher confidence to classes that are challenging for detection process. Subsequently, weighted voting aggregates these predictions, ensuring that each classifier's strengths are utilized. The final decision is based on the class with the highest aggregated score. Meanwhile the ensemble model's parameters are tuned using the Bayesian optimization process.

Algorithm 1

Input : Raw Simulated IoT sensor data

Output: Multi class fault detection (Voltage, Temperature, Speed and Vibration)

Begin

Data preprocessing

Standardization - Z score scaling

$$sf' = (of - \text{mean}(of)) / (\text{sd}(of))$$

Removal of Non numerical values

Categorical Encoding

Timestamp Parsing

Feature Selection

Stage-Wise Feature Selection

Voltage Vs Timestamp

Speed Vs Timestamp

Temperature Vs Timestamp

Vibration Vs Timestamp

Depending on factors: Very low, low, normal & high

Feature Extraction:

Optimized Robust PCA

$\min_{LM, SM} LM _* + \lambda SM _1$	
Stagewise Sample Extraction	
Voltage Vs Timestamp (LM ₁ , SM ₁)	
Speed Vs Timestamp (LM ₂ , SM ₂)	
Temperature Vs Timestamp (LM ₃ , SM ₃)	
Vibration Vs Timestamp (LM ₄ , SM ₄)	
Optimized Ensemble Machine Learning based fault detection model:	
Ensemble model (KNN, Adaboost)	
Bayesian Optimization for IoT enabled Machinery	
Return Optimal parameters of ensemble model	
Performance computation:	
Accuracy, Detection rate, Precision and	
FNR	
End	

3.6. Optimization Process - Bayesian optimization

Hyper parameter tuning plays a critical role in improving the performance of machine learning models. Manual or grid-based tuning is inefficient and time-consuming whilst the Bayesian Optimization offers a systematic and efficient way to identify optimal hyper parameters by modelling the objective function and iteratively refining the search space. The working procedure of the Bayesian Optimization is shown in flowchart of Fig.3

Bayesian Optimization involves constructing a probabilistic model, typically a Gaussian Process (GP), to approximate the objective function. Based on this model, the algorithm identifies hyper parameter configurations that yields optimal results. The parameters of KNN and Adaboost considered for optimization process are tabulated in table 2 (i) Number of Neighbours, (ii) Distance Metric - Manhattan , (iii) Learning rate (), (iv) Estimators and (v) Depth of the tree. The iteration process ends once the fitness function is reached.

Table 2 Bayesian Parameters & its Values

Parameters	Values
Number of neighbours	$n = 4$
Distance Metric (Manhattan distance)	$x_{s1} - x_{s2} + y_{p1} - y_{p2}$
Learning rate	$\lambda_i = 0.001$
Estimators	$n_1 \text{ estimator} = 300$
Depth of the tree	$D_p = 3$

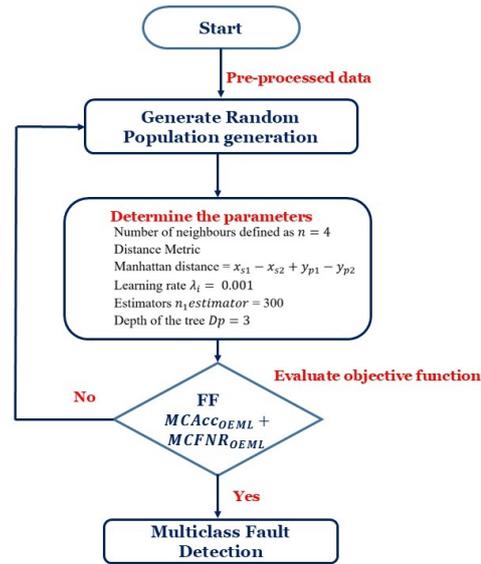


Fig.3 Working of Bayesian optimization-based ensemble learning model

The following formula (3) specifies the objective function considered for multiclass fault detection

$$\text{Fitness Function} = \text{MCAcc}_{\text{OEML}} + \text{MCFNR}_{\text{OEML}} \quad (3)$$

Where the $\text{MCAcc}_{\text{OEML}}$ denotes the aggregated accuracy of multi class fault detection, $\text{MCFNR}_{\text{OEML}}$ mentions the aggregated False Negative Rate. These two metrics plays the significant role in fault detection process and upholds the learning model's performance; it represents the accurate detection and the rate of negative samples which are wrongly predicted.

4. Experimental Details:

This section demonstrates the analysis carried for fault detection in IoT enabled machinery. The model is evaluated using simulated IoT enabled machinery dataset. The following subsection provides the detailed data description. The proposed model encompasses data collection followed by stage wise feature selectors which is processed by imposing the timestamp feature which is common to correlate with all the features and the model produce feature subsets based on the timestamp. Then, Optimization concept is implemented for feature extraction and detection method. Optimal Features are extracted from the R-PCA method using the tuning concept. λ Parameter is tuned to manage the low rank and sparse PCA components. Also, the ensemble models are optimised using the Bayesian optimization method by tuning the specific parameters of learning model which enhances the performance and produce the accurate results. The experiments analyse few performance metrics while examining computational efficiency, interpretability, and feature importance. The results provide insights into model selection and optimization strategies, emphasizing the trade-offs between complexity, accuracy, and application requirements.

4.1. Experimental setup:

The proposed optimized Ensemble learning model was implemented in python 3.5 with the TensorFlow framework and keras library was used for evaluation process. The experimentation was conducted on labelled Industry based IoT enabled machinery and performed on intel core i7 processor with 16 GB RAM running Windows 11 OS.

4.2. Dataset Description

The dataset comprises sensor data recorded using NI LabVIEW programming software, which provides a flexible and robust platform for real-time data acquisition, analysis, and visualization. Each entry in the dataset is timestamped with the current date and time, allowing for historical analysis and the identification of trends critical for predicting potential machine failures. The dataset includes 3608 samples with seven columns: Date, Time, Voltage, Speed, Vibration, Temperature, and Pressure. The voltage readings range from 220 to 250 volts, capturing fluctuations in electrical supply. Speed measurements span from 1000 to 3000 m/s, reflecting the operational dynamics of the machine. Vibration data, ranging between 300 and 600 m/s^2 , provides insights into mechanical stability and potential anomalies. Temperature values range from 15 to 55 degrees Celsius, enabling monitoring of thermal conditions that may indicate overheating or suboptimal operating environments. Pressure values fall between 80 and 160 Pascals, highlighting variations that may impact system performance. This dataset presents a comprehensive log of machine parameters, facilitating long-term performance analysis and enabling engineers to identify gradual trends or faults in machineries that may lead to future malfunctions.

4.3. Data pre-processing & Feature Engineering

The collected data is pre-processed before training the model for fault detection in IoT enabled machinery. The process included to pre-process the data are data cleaning, integration and transformation. The standardization of z-score scaling is applied to scale the values of the attributes of voltage, pressure, speed, temperature and vibration. For conversion of date and time of dataset as single column, the method of time parsing is deployed. Fig. 4 represents the time parsing in the IoT enabled machinery data. Due to the absences of specifications especially in case of analysing the fault in the IoT enabled machinery, the categorical labelling is used to label the categories are very low, low, medium, high and very high. As per data cleaning, the dataset are cleaned by removing all unwanted and non-numerical values. Once, the data are pre-processed, the feature engineering techniques are employed to the stage wise feature selector using feature selection technique and extract the essential samples through feature extraction.

VOLTAGE (VOLTS)	PRESSURE (Pascal)	SPEED (m/s)	TEMP (Deg C)	VIBRATION (m/s^2)	TIMESTAMP
230	85	1500	45	403	2021-02-03 15:43:32
220	85	1500	38	405	2021-02-03 15:43:32
220	85	1600	38	403	2021-02-03 15:43:32
225	85	1600	35	504	2021-02-03 15:43:32
230	85	1000	36	510	2021-02-03 15:43:32

Fig.4 Sample data after time parsing

Feature selection for multi-correlated features, the stage-wise feature selection method is presented by combination of timestamp with all the attributes of dataset are voltage, speed, temperature and vibration. The obtained stage wise feature shows the subset of different stages that defines the sensitive operational mechanisms of IoT sensors.

Stage 1: Voltage Vs Timestamp

The voltage fluctuations are noted from 220V to 250 V where 220 V specifies the very low, 220 to 230 turns low volt, 230 to 240 turns normal ,240 to 250 volt is high and when the voltage goes greater than or equal to 250 volts, it reaches high Volt. Thus, samples which covers except these ranges are selected based on timestamp. The well working state of the machinery are coming under normal and low where normal volt is comfortable compared to low.

Stage 2: Speed Vs Timestamp

The speed (m/sec) of a IoT sensors are marked from 1000 to 2000 meters per second. It also classified in four categories from very low to high. The speed which goes less than 1000 will makes the IoT sensor to malfunction. Similarly, the speed less than 1500 are marked as low speed that slowdown the entire process of the IoT sensors. The normal speed for the machinery with IoT sensors from 1500 to 2000 that practices the machinery with balanced speed and the speed of machinery when works greater than 2000 that shows the IoT sensor-based machinery are working out of control. So, the machinery should be worked in two cases are normal and low , where normal is possible however, low speed can be ignored.

Stage 3: Temperature Vs Timestamp

The temperature for the working state of IoT enabled machinery are defined in four different cases.

Case 1: very low- A temperature lies less than 15 degrees Celsius

Case 2: low-A temperature lies between 15 to 25 degrees Celsius

Case 3: Normal - A temperature lies from 25 to 45 degrees Celsius

Case 4: High-A temperature from 45 to 55 & greater than 55 degrees Celsius

When a temperature reaches very low or high state leads to thermal stress or overheating to the machinery that cause failure in machinery.

Stage 4: Vibrations Vs Timestamp

It concentrates on monitoring oscillatory motion using sensors to predict the deviations from normal behaviour of the machinery. Normal state of vibrations lies from 400 to 500 that notify the steady state of IoT enabled sensors. However, the rest of the three cases are very low has less than 300, low lies between 300 to 400 and high 500 to 600.Its analysis when a specific fault occurred in the machinery. Thus, stage -wise feature selection method represents the importance of the attributes with timestamp and the four different cases for each attribute, among those cases, the monitoring the three cases (very low, low and high) in mandatory because it leads to fault the machinery in various chances. Thus, all the stage-wise feature selector are shown in Fig.5

Timestamp	VOLTAGE (VOLTS)	Timestamp	SPEED (m/s)	Timestamp	TEMP (Deg C)	Timestamp	VIBRATION (m/S2)
02-03-2021 15:43:32	230	02-03-2021 15:43:32	1500	02-03-2021 15:43:32	45	02-03-2021 15:43:32	403
02-03-2021 15:43:32	220	02-03-2021 15:43:32	1500	02-03-2021 15:43:32	38	02-03-2021 15:43:32	405
02-03-2021 15:43:32	220	02-03-2021 15:43:32	1600	02-03-2021 15:43:32	38	02-03-2021 15:43:32	403
02-03-2021 15:43:32	225	02-03-2021 15:43:32	1600	02-03-2021 15:43:32	35	02-03-2021 15:43:32	504
02-03-2021 15:43:32	230	02-03-2021 15:43:32	1000	02-03-2021 15:43:32	36	02-03-2021 15:43:32	510
02-03-2021 15:43:32	230	02-03-2021 15:43:32	1010	02-03-2021 15:43:32	25	02-03-2021 15:43:32	550
02-03-2021 15:43:32	225	02-03-2021 15:43:32	1200	02-03-2021 15:43:32	29	02-03-2021 15:43:32	528
02-03-2021 15:43:32	240	02-03-2021 15:43:32	1700	02-03-2021 15:43:32	42	02-03-2021 15:43:32	540
02-03-2021 15:43:32	224	02-03-2021 15:43:32	1900	02-03-2021 15:43:32	40	02-03-2021 15:43:32	560
02-03-2021 15:43:32	200	02-03-2021 15:43:32	2000	02-03-2021 15:43:32	40	02-03-2021 15:43:32	580
02-03-2021 15:43:32	220	02-03-2021 15:43:32	3000	02-03-2021 15:43:32	45	02-03-2021 15:43:32	600
02-03-2021 15:43:32	210	02-03-2021 15:43:32	3500	02-03-2021 15:43:32	50	02-03-2021 15:43:32	540
02-03-2021 15:43:32	220	02-03-2021 15:43:32	500	02-03-2021 15:43:32	50	02-03-2021 15:43:32	520
02-03-2021 15:43:32	250	02-03-2021 15:43:32	500	02-03-2021 15:43:32	45	02-03-2021 15:43:32	500
02-03-2021 15:43:32	250	02-03-2021 15:43:32	0	02-03-2021 15:43:32	44	02-03-2021 15:43:32	480
02-03-2021 15:43:32	0	02-03-2021 15:43:32	0	02-03-2021 15:43:32	38	02-03-2021 15:43:32	455
02-03-2021 15:43:32	0	02-03-2021 15:43:32	2427	02-03-2021 15:43:32	29	02-03-2021 15:43:32	462
02-03-2021 15:43:32	233	02-03-2021 15:43:32	2427	02-03-2021 15:43:32	26	02-03-2021 15:43:32	445
02-03-2021 15:43:32	233	02-03-2021 15:43:32	2427	02-03-2021 15:43:32	26	02-03-2021 15:43:32	436
02-03-2021 15:43:32	229	02-03-2021 15:43:32	1458	02-03-2021 15:43:32	30.3	02-03-2021 15:43:32	412

Fig. 5 Sample Stage-wise Feature Selectors**4.4 Comparison on effectiveness of the proposed stage-wise feature selector model**

The proposed stage-wise feature selection model is substantially compared with the baseline models, such as PCA, Mutual Information, Relief, and Recursive Feature Elimination (RFE). The comparison is performed based on the identical experimental valuation using the same ensemble classifiers and valuation metrics.

Models	Accuracy	Precision	Detection rate	FNR
PCA	0.921	0.912	0.934	0.12
Mutual Information	0.957	0.941	0.956	0.15
ReliefF	0.964	0.954	0.948	0.10
RFE	0.914	0.897	0.914	0.17
Proposed Feature Selection Model	0.985	0.984	0.988	0.02

Table 3 clearly captures the quantitative comparison analysis of the proposed stage-wise feature selection model. To ensure the fairness of all the models, the evaluation is based on the same classifier model, of which the proposed model outperforms other models with the scores 0.985 accuracy, 0.984 precision, 0.988 Detection rate and 0.02 FNR. Though ReliefF exhibits better accuracy of 0.964, it is 2% lesser than the proposed model. The method of feature extraction used for fault detection in IoT-enabled machinery is robust principal component analysis (RPCA). Table 3 presents the detailed view of the parameters taken for training the learning model. To optimize the r-PCA, the learning parameter "Lamda λ " is tuned until the model extracts the relevant feature samples. λ controls the trade-off between the low-rank representation of data and the sparse noise matrix during dimensionality reduction. The kernel function and the number of components specifically denote the model's effectiveness.

Table 3 RPCA Components & their Description

Component	Parameter	Value	Optimization Model
RPCA	λ	Optimal values are used after training	Domain Knowledge based optimization
	Number of Components	Automatically determined based on explained variance	
	Kernel	Radial Basis Function (RBF)	

Given the machinery dataset with attributes such as timestamp, voltage, speed, temperature, and vibration, R-PCA is applied to each stage-wise subset to extract meaningful features:

Stage 1: Voltage Vs Timestamp: The temporal voltage patterns often reveal deviations due to electrical instability. RPCA isolates the low-rank matrix LM_1 representing the nominal voltage trends over time, while SM_1 captures sudden voltage spikes or drops, which may indicate electrical faults

Stage 2: Speed Vs Timestamp: Variations in machinery speed over time can signal performance degradation or mechanical issues. RPCA applied to this subset extracts M_2 , which encapsulates consistent operational speed patterns, and SM_2 identifying abnormal fluctuations.

Stage 3: Temperature Vs Timestamp: Temperature readings are influenced by both environmental and operational conditions. The RPCA decomposition yields M_3 , representing typical temperature evolution, and SM_3 , isolating thermal anomalies like overheating.

Stage 4: Vibration Vs Timestamp: Vibration data is critical for diagnosing mechanical wear or alignment issues. By applying RPCA, LM_4 captures the inherent vibration characteristics, while SM_4 highlights unusual patterns associated with faults.

In Fig. 6, the blue plots from the figure denotes the Standard PCA, which projects the data onto principal components (PCs) that maximize variance. While it reveals the structure of the dataset, it is sensitive to outliers and noise, as seen by the scattered points deviating from the core data structure. In fault detection, this sensitivity may cause false alarms due to the influence of outliers or irrelevant noise. While the green plots and red plots refers to the representation of R-PCA. It separates the data into a low-rank component (core structure) and a sparse component (anomalies and noise). The plot focuses on the low-rank component, capturing the underlying normal behavior of the system while suppressing noise. The sparse component highlights noise and anomalies (data points that deviate significantly from the normal pattern). Figure 5 demonstrate how r-PCA outperforms standard PCA by providing a clearer separation of normal and anomalous data, making it highly suitable for complex industrial systems.

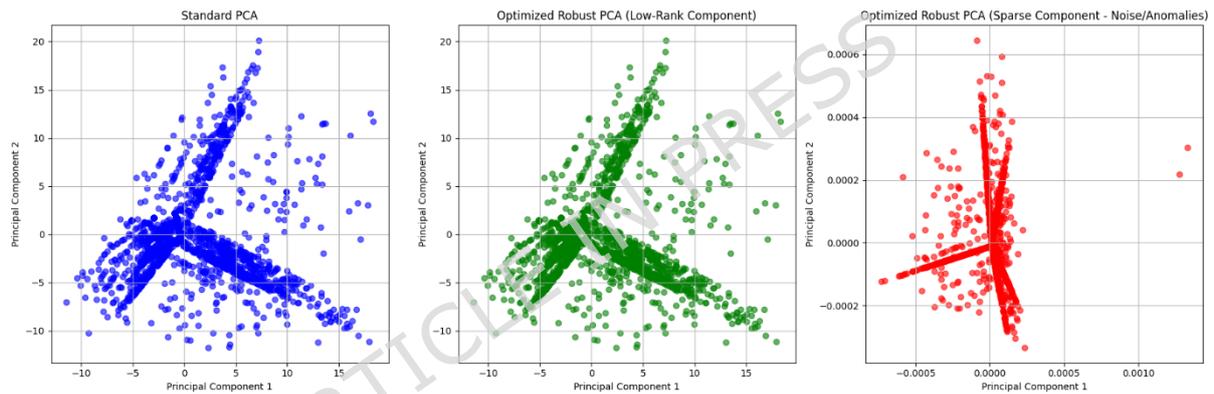


Fig.6 Scattered plot of Standard PCA, optimized RPCS (Low rank), Optimized RPCA(Sparse)

The extracted low-rank matrices LM_1, LM_2, LM_3, LM_4 serve as compact representations of the original data, aiding the fault detection model to focus on the most informative aspects of the machinery's operational states. Simultaneously, the sparse matrices SM_1, SM_2, SM_3, SM_4 provide actionable insights into anomalies, which are essential for pre-emptive maintenance. Optimization-based RPCA further refines this process by adaptively tuning λ to balance the contributions of LM and SM based on the dataset's specific characteristics. By integrating optimization-based RPCA into the feature extraction pipeline, enhances the reliability and precision of subsequent fault detection tasks, ensuring robust diagnostics and minimizing downtime in IoT environment.

To ensure the quantitative stability of the proposed RPCA method, the decomposition of the λ is varied with the different values stating from 0.005 to 0.050. Table 5 presents the detailed quantitative analysis based on the metrics reconstruction error, sparsity ratio, Detection rate and accuracy.

Reconstruction error is defines with the normalization form of frobenius where it is expressed as Eqn. 4

$$(R) = \|X - L - S\|_F / (\|X\|_F) \quad (4)$$

X represents the data matrix, L represents the low rank component, and Sparse Component S . F represents the Frobenius norm. The reconstruction error is evaluated in the form that low scores define the more accurate and stable RPCA decomposition form. The sparsity ratio will measure the proportion of non-zero elements and reflects which anomalies or fault information are isolated. Eqn.5

$$S (\%) = \frac{\|s\|_0}{m*n} * 100 \quad (5)$$

$\|s\|_0$ represents the pseudo normalization with non-zero elements and $m * n$ represents the number of rows and columns present in the sparse matrix.

λ Values	Reconstruction error	Sparsity Ratio	Detection Rate	Accuracy
0.005	0.241	9.4	0.917	0.915
0.010	0.213	8.7	0.924	0.927
0.015	0.187	8.4	0.964	0.935
0.020	0.176	7.3	0.978	0.958
0.025 (Optimized)	0.102	6.5	0.988	0.985
0.030	0.145	7.9	0.964	0.961
0.035	0.167	8.1	0.954	0.958
0.040	0.154	8.4	0.941	0.942
0.045	0.165	8.8	0.924	0.921
0.050	0.197	9.1	0.912	0.901

The results indicate that the λ parameter heavily relies on the RPCA behaviour. If the value of λ parameter is less then it leads to high reconstruction errors with insufficient sparsity ratio values. However, if the value is high, then it leads to less reconstruction error and fluctuations in the sparsity with low accuracy and low detection rate. Whereas, the optimized value of 0.025, achieves the better value in all the aspects with higher accuracy of 0.985, detection rate with 0.988, balanced sparsity ratio as 6.5 and minimal reconstruction error with 0.102. Hence the presence of the parameter with optimised value remains the marginal, following decomposition stability and robustness.

4.4. Fault Detection in IoT enabled machinery using Optimized Ensemble Machine Learning

The ensemble machine learning model is the combination of KNN and Adaboost which are optimized by Bayesian optimization for fine tuning the learning rate of the ensemble learning model. While designing the ensemble learning model the parameters shown in table 4, the higher γ can smooth out noise but risks overgeneralizing, while a lower γ increases sensitivity to local patterns. Setting $\gamma = 4$ ensures that the classification balances noise reduction and sensitivity, suitable for detecting subtle fault signatures in machinery. The Manhattan distance emphasizes horizontal and vertical movements, which suits high-dimensional IoT sensor data where correlations between features might not follow Euclidean behaviour. It provides a simple yet effective metric for fault classification, especially for sensor readings like vibration, speed, and temperature. A small learning rate ($\lambda_i = 0.0001$) ensures that weak learners do not overpower the ensemble model, allowing gradual improvements in performance. It prevents overfitting by spreading learning across multiple iterations, making the model robust to unseen data. Setting $n1 = 300$ ensures the ensemble model has enough complexity to capture subtle fault patterns while avoiding overfitting. A larger number of estimators improves

classification accuracy for multi-class problems, such as distinguishing between vibration, speed, voltage, and temperature faults. With $dp = 3$ the decision trees remain simple, reducing the risk of overfitting and improving computational efficiency. Shallow trees focus on identifying strong, generalized fault indicators rather than overfitting to minor variations in the data. By tuning these parameters using Bayesian optimization, the model is well equipped and produced better results in all aspects. By computing the Mean Squared Error (MSE), the loss occurs 0.02 for testing and 0.03 for training is achieved for 100 epochs. In Fig. 6, blue denotes the training loss and yellow represents the testing loss decrease steadily as the number of epochs increases, demonstrating that the model is learning effectively. The loss reduction is significant during the initial epochs, with diminishing returns as training progresses. The testing loss is slightly lower than the training loss for most epochs. This indicates that the model has been regularized properly or benefits from a well-generalized dataset, prevents overfitting. The two curves converge toward the end, with minimal differences in loss values, which is a sign of balanced training. Both training and testing losses stabilize near 0.05 to 0.1 after 80 epochs, indicating that the model achieves low errors for both training and unseen data.

Table 4 Optimized Ensemble Learning model Parameters & its Descriptions

Component	Parameter	Value	Optimization Model
K-Nearest Neighbours (KNN)	Number of Neighbours (n)	4	Bayesian Optimization
	Distance Metric	Manhattan Distance	
	Weight Function	Uniform	
Adaboost	Learning rate λ_i	0.0001	
	Number of Estimators n1	300	
	Max. No. of. Depth	3	
	Loss Function	Modified Huber Loss	

The optimized ensemble learning model have reached accuracy of 0.987 for 100 epochs. Fig.7, represents both training and testing accuracy increase gradually from the start of training, indicates effective model learning. The curves exhibit a steady improvement, which denotes that the model is progressively capturing the underlying patterns in the data. The testing accuracy curve slightly surpasses the training accuracy curve, which is an uncommon but possible occurrence due to regularization, early stopping, or other factors minimizing overfitting. The closeness of the two curves implies that the model generalizes well to unseen data. The initial accuracy starts around 88.1 for training and 0.90 for testing, improving to nearly 0.98 for both by the 100 epochs. The smooth curvature indicates that learning is consistent without significant fluctuations, suggesting a well-tuned model and training process.

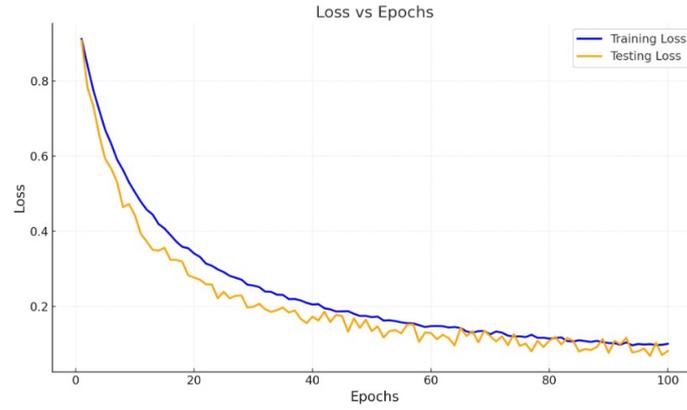


Fig.6 Loss Vs Epoch of proposed Model

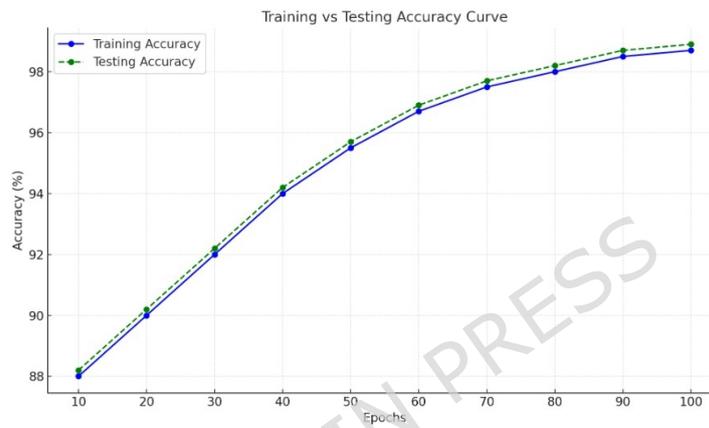


Fig.7 Accuracy Vs Epoch of Proposed Model

5. Results and Discussion

The performance of the proposed optimized ensemble learning model is evaluated through metrics are accuracy, precision, detection rate and false negative rate given in Eqn. (6) to (9)

$$\text{Accuracy} = \frac{TP+TN}{(TP+TN+FP+FN)} \quad (6)$$

$$\text{Precision} = \frac{TP}{(TP+FP)} \quad (7)$$

$$\text{Detection rate} = \frac{TP}{(TP+FN)} \quad (8)$$

$$\text{FNR} = \frac{FN}{(FN+TP)} \quad (9)$$

To ensure the effectiveness of the proposed method and all the comparative models are evaluated using the stratified k-fold cross validation was employed. Precisely, 10-fold cross validation is performed where the dataset is divided into 10 folds and the originals classes are maintained in each folds. This guarantees that the samples are given proportionally to both the training and testing sets. In each reiteration, the dataset was rationalized and re-partitioned formerly cross-validation, causing in multiple independent valuation runs. The concluding test results correspond to the mean and standard deviation of these metrics across all folds and repetitions. This evaluation strategy suggestively moderates modification caused by random sampling effects and provides a more robust estimate of the model's generalization proficiency, predominantly for datasets of modest size. This comprehensive validation procedure strengthens the reliability of the performance metrics and supports the practical applicability of the proposed fault detection framework.

5.1. Impact of Accuracy on IoT enabled machinery dataset

Accuracy represents the proportion of correct predictions (both true positives and true negatives) relative to the total predictions, playing a critical role in how effectively a system interprets sensor data. Considering the four key factors—temperature, vibration, voltage, and speed—their impact on accuracy is analysed as follows:

- (i) **Temperature:** Variations in temperature leads to sensor drift or inaccurate readings. Extreme temperatures may alter sensor sensitivity or calibration, thereby reducing system accuracy. Ensuring a stable thermal environment helps maintain sensor integrity and improves accuracy by minimizing temperature-induced errors.
- (ii) **Vibration:** High levels of vibration will distort sensor outputs, particularly for precision instruments. Excessive vibrations introduce noise, which compromises the reliability of the data and decreases overall accuracy. Implementing vibration-dampening measures can mitigate this effect and enhance sensor performance.
- (iii) **Voltage:** Fluctuations in voltage disrupts sensor performance, leading to errors in signal transmission or data acquisition. These disruptions undermine system accuracy. Maintaining a stable and consistent voltage supply enhances sensor reliability, directly contributing to improved accuracy.
- (iv) **Speed:** In systems that measure rotational or linear speed, inaccuracies often arise due to miscalibration or noise in the measurements. These errors propagate through the model, negatively impacting overall system accuracy. Proper calibration and filtering techniques are essential to ensure precise speed measurements and maintain accuracy. By addressing these factors, system accuracy is significantly improved with the value 0.984 and ensures reliable interpretation of sensor data and overall system performance.

5.2. Impact of Detection Rate on IoT Enabled machinery dataset

The detection rate, also known as sensitivity, represents the proportion of actual positives accurately identified. It plays a vital role in fault prediction. The influence of detection rate is analysed across the following key factors:

- (i) **Temperature:** Variations in temperature, such as overheating in motors is an early indicator of potential faults. Sensors must be sufficiently sensitive to detect even minor deviations, thereby improving the detection rate and enabling timely interventions.
- (ii) **Vibration:** Abnormal vibrations are often precursors to mechanical failures. Achieving a high detection rate is critical to promptly identify these irregularities. Insufficient sensitivity may result in missed critical warning signs, compromising system reliability.
- (iii) **Voltage:** Voltage fluctuations, including drops or spikes, often indicate power issues that could lead to equipment damage or failure. A high detection rate ensures that such abnormalities are quickly identified, allowing corrective actions to prevent system disruptions
- (iv) **Speed:** Sudden changes in rotational or linear speed often signal malfunctions in the system. A high detection rate of 0.987 enables the accurate identification of faults and ensures timely responses to prevent further damage or operational inefficiencies.

5.3. Impact of Precision on IoT enabled machinery dataset

Precision measures the proportion of correctly identified positives out of all predicted positives, playing a critical role in minimizing false alarms and ensuring system trustworthiness. Its importance is examined across key factors as follows:

- (i) **Temperature:** Minor, non-critical temperature variations can trigger false alarms, disrupting operations. Sensors with high precision effectively differentiate

between normal fluctuations and genuine faults, reducing unnecessary interventions.

- (ii) **Vibration:** Precision is crucial in detecting critical vibration thresholds while ignoring benign variations. Excessive false alarms caused by low precision leads to unnecessary shutdowns or corrective actions, reducing system efficiency.
- (iii) **Voltage:** Voltage sensors need to accurately identify problematic fluctuations without being influenced by transient variations also high precision ensures smooth operations by preventing false positives that could unnecessarily disrupt processes
- (iv) **Speed:** In systems measuring speed, high precision ensures that only significant deviations are flagged. This reduces the risk of false alarms, minimizing unnecessary corrective actions and maintaining system stability. Precision value of 0.981 in detecting faults avoids disruptions caused by false positives and improves the overall efficiency.

5.4. Impact of False Negative Rate on IoT enabled machinery dataset

False negative rate (FNR) measures the proportion of actual positives that go undetected by the system. Maintaining a low FNR is essential for ensuring safety and reliability. Its impact across key parameters is as follows:

- (i) **Temperature:** Missing critical temperature changes, such as overheating, can result in severe failures or damage. A low FNR ensures the system responds promptly to such changes, preventing catastrophic outcomes.
- (ii) **Vibration:** Undetected abnormal vibrations can significantly increase the likelihood of mechanical failures. A low FNR ensures these issues are identified early, allowing timely interventions to mitigate risks.
- (iii) **Voltage:** Failing to detect voltage anomalies, such as spikes or drops, damage connected devices or disrupt operations. Minimizing the FNR safeguards the system against power-related issues and ensures smooth functionality.
- (iv) **Speed:** Undetected anomalies in speed measurements can cause inefficiencies or accelerated system wear. Reduced FNR rate of 0.02 enhances the system's ability to identify such faults.

5.5. Comparison of proposed model with Existing ML models

The proposed optimized ensemble learning model is compared with other machine learning model such as Decision Tree [39], Random Forest [40], Navie Bayes [41] , K-Nearest Neighbour and Decision Stump [42] depending on the performance metrics are accuracy, precision, detection rate and false negative rate. The proposed model has identified the fault in the IoT enabled machinery and classified those faults as multiclass (class 0 to class 4) shown in table 5.

Table 5 Multiclass w.r.t Dataset and its Description

Multiclass	Descriptions
Class 0	Fluctuation in voltage
Class 1	High or low Vibrations
Class 2	Increase or decreases in speed range
Class 3	Change temperature level

Table 6 Comparison of Proposed model with existing models based on Accuracy

Learning models	Accuracy			
	Class 0	Class 1	Class 2	Class 3
Decision Tree	0.932	0.937	0.921	0.927

Random Forest	0.921	0.929	0.924	0.927
Navie Bayes	0.911	0.915	0.920	0.919
KNN	0.925	0.921	0.926	0.930
Decision Stump	0.945	0.947	0.951	0.954
Ensemble model	0.984	0.982	0.986	0.988

In table 6, the accuracy of the proposed optimized ensemble learning model is high for all class 0 to 4 compared to other models. The proposed model has high accuracy for class 0 (0.984), class 1 (0.982), class 2(0.986) and class 3 (0.988) compared to all other learning models. The decision tree has low accuracy (0.932 to 0.927) due to the high variance that leads to the overfitting. Due to the complexity of the random forest model, it occurs less accuracy (0.921 to 0.927). In case of Navie Bayes, the accuracy (0.911 to 0.919) because of the independent features which are not highly correlated. The accuracy of KNN covers from (0.925 to 0.930), however the choosing the neighbour nodes “k” whether its high or low leads to risk in fault identification. The last model of Decision stump has slight high accuracy (0.945 to 0.954) compared to all models, but low to the proposed model because, it concentrates only on single feature.

Table 7 Comparison of Proposed model with existing models based on Precision

Precision				
Learning models	Class 0	Class 1	Class 2	Class 3
Decision Tree	0.926	0.921	0.927	0.920
Random Forest	0.909	0.901	0.908	0.911
Navie Bayes	0.932	0.937	0.939	0.940
KNN	0.942	0.945	0.947	0.949
Decision Stump	0.956	0.962	0.963	0.968
Ensemble model	0.981	0.983	0.985	0.987

In table 7, the precision of the proposed optimized ensemble learning model has obtained sequential improvement from class 0 to class 1. Class 3 (0.987) has higher precision compared to class 0 (0.981), class 1(0.983) and class 2(0.985) obtained from the proposed ensemble model. The decision tree has lower precision from (0.926 to 0.920) because of complex decision boundaries. In random forest, the less precision occurs from (0.909 to 0.911) due to limited feature. In this data, the features are dominant each other that makes Navie bayes to have a less precision value from (0.932 to 0.940). Usually, KNN is less sensitive to identify the outliers that has less precision value from (0.942 to 0.968). The decision stump also has improved precision value from (0.956 to 0.968) but lesser than proposed model. Thus, the precision of the proposed model ensures the detection of faults in IoT enabled machinery.

Table 8 Comparison of Proposed model with existing models based on Detection Rate

Detection Rate				
Learning models	Class 0	Class 1	Class 2	Class 3
Decision Tree	0.943	0.932	0.938	0.939
Random Forest	0.927	0.922	0.932	0.929
Navie Bayes	0.943	0.947	0.949	0.945
KNN	0.953	0.955	0.957	0.959
Decision Stump	0.961	0.965	0.967	0.969

Ensemble model	0.987	0.988	0.989	0.988
-----------------------	--------------	--------------	--------------	--------------

In general, detection rate proves the proposed model for the fault detection in IoT enabled machinery from class 0 (0.987), class 1(0.988), class 2(0.989) and class 3(0.988). Similar to accuracy and precision, the detection rate of the proposed model has higher value compared to decision tree, random forest, naive bayes, KNN, decision stump is shown in table 8. The false negative rate of the proposed model for classes (0.02, 0.03) defines assumptions of actual faults identification in IoT enabled machinery. Compared to all learning model, the proposed model has less FNR value shown in table 9.

Table 9 Comparison of Proposed model with existing models based on False Negative Rate (FNR)

False Negative Rate				
Learning models	Class 0	Class 1	Class 2	Class 3
Decision Tree	0.07	0.09	0.07	0.06
Random Forest	0.08	0.06	0.06	0.07
Naive Bayes	0.06	0.08	0.08	0.08
KNN	0.05	0.04	0.05	0.04
Decision Stump	0.04	0.05	0.07	0.06
Ensemble model	0.02	0.03	0.03	0.02

Table 10: Comparative Results of Assessed Machine Learning Models

Models	Accuracy	Precision	Detection rate	FNR
KNN	0.934	0.928	0.941	0.059
MLP	0.951	0.946	0.954	0.046
AdaBoost	0.962	0.958	0.965	0.035
XGBoost	0.971	0.968	0.973	0.027
ANN	0.948	0.942	0.952	0.048
DNN	0.968	0.964	0.971	0.029
Ensemble Model	0.985	0.984	0.988	0.02

The comparative results in the Table 10. Shows the distinct performance hierarchy for the assessed machine learning models. KNN and Shallow ANN model achieved higher accuracy however the false negative rate for both relies high (0.059 for KNN and 0.018 for ANN) when compared with other models, this shows that the model is not sufficient to detect the subtle fault patterns. MLP and AdaBoost models presents improved performance in terms of accuracy, precision, detection rate and FNR due to its non-linear relationship ability and ensemble decision making process. XGBoost and DNN effectively captures the complex feature interactions, despite all these the proposed models outperform the rest of the models in all the metrics with the score of 0.985 (accuracy), 0.984 (Precision), 0.988 (Detection Rate) and 0.02 (FNR). The significant result of FNR for the proposed model exhibits that it is crucial for industrial fault detection, as missed fault even leads to machine damage, machinery breakdown and fault analysis. Hence the

proposed model ensures that the lowest FNR leads to increased productivity with less false rate.

Discussion on Statistical Significance and Confidence Interval analysis

To ensure that the performance of the proposed model is not quantifiable to certain data and to prove its robustness, the confidence interval analysis was carried out with the mean \pm standard deviation in terms of accuracy and detection rate. Table 11 presents the detailed comparative analysis of confidence interval analysis with the score of 95%. Mean and standard deviations are computed over various repeated 10-fold cross validation process. From which it is observed that the proposed models exhibit the lowest variance and slenderest confidence intervals to disclose the superior robustness and generalization levels.

Table 11: Comparative Results of Assessed Machine Learning Models

Models	Accuracy	Accuracy (Mean \pm SD)	Detection Rate	95% CI	Detection rate (Mean \pm SD)	95% CI	Statistical Significance
KNN	$3.2 * 10^{-6}$	0.934 ± 0.012	$2.7 * 10^{-6}$	[0.910, 0.958]	0.941 ± 0.011	[0.919, 0.963]	Significant
MLP	$1.2 * 10^{-5}$	0.951 ± 0.010	$8.6 * 10^{-6}$	[0.931, 0.971]	0.954 ± 0.009	[0.936, 0.972]	Significant
AdaBoost	$4.3 * 10^{-5}$	0.962 ± 0.009	$3.9 * 10^{-5}$	[0.944, 0.980]	0.965 ± 0.008	[0.949, 0.981]	Significant
XGBoost	$6.8 * 10^{-4}$	0.971 ± 0.008	$5.1 * 10^{-4}$	[0.955, 0.987]	0.973 ± 0.007	[0.959, 0.987]	Significant
ANN	$2.4 * 10^{-3}$	0.948 ± 0.011	$1.9 * 10^{-5}$	[0.926, 0.970]	0.952 ± 0.010	[0.932, 0.972]	Significant
DNN	$9.7 * 10^{-4}$	0.968 ± 0.008	$7.8 * 10^{-4}$	[0.952, 0.984]	0.971 ± 0.007	[0.957, 0.985]	Significant
Ensemble Model	$9.8 * 10^{-4}$	0.985 ± 0.004	$7.4 * 10^{-4}$	[0.977, 0.993]	0.988 ± 0.003	[0.982, 0.994]	Significant

The paired t-test was evaluated at a confidence level of 95% with the significance value $\alpha = 0.05$, using the 10-fold cross validation. From the table [11] it is evident that all the models are below the significance value and it is computationally significant in terms of accuracy and detection rate which holds the primary role in analysing the machinery faults. The findings confirm that the improvements in performance are not attributed to random fluctuations but are a result of the proposed model's optimized RPCA-based feature extraction, sequential feature selection, and Bayesian-tuned ensemble learning approach.

In Industrial environments powered by IoT, where edge devices must function within stringent limits on processing power, memory, and energy, computational efficiency is as vital as diagnostic precision for fault detection systems. The RPCA-based optimized ensemble framework introduced here is crafted to emphasize lightweight computation and facilitate real-time application. This method diverges from a standard lightweight deep learning baseline by significantly lowering training complexity through the use of

matrix decomposition and shallow learning models, thus bypassing the need for iterative gradient-based optimization across numerous network layers. The model also achieves reduced inference latency by forgoing deep hierarchical feature transformations, instead utilizing low-overhead distance calculations and ensemble voting. Furthermore, the memory requirements of the proposed framework are kept minimal, as it only retains the RPCA components, selected feature subsets, and classifier parameters, unlike deep learning models that necessitate large weight matrices and intermediate activations. These features render the proposed method exceptionally apt for ongoing, real-time fault monitoring in industrial settings.

6. Conclusion

The experimental results of the proposed optimized ensemble learning model demonstrate the model's efficacy in fault detection, achieving remarkable performance metrics with an accuracy of 0.984, precision of 0.981, a detection rate of 0.987, and a low false negative rate of 0.02. These metrics underscore the robustness and reliability of the proposed framework, particularly in identifying faults accurately by ensuring the high detection with minimal false negative rate. By addressing the challenges of noisy data and high-dimensional sensor inputs, the optimized robust PCA method proves the quality of feature extraction, while the ensemble model provided the adaptability and predictive strength required for real-world fault diagnosis in industrial systems. Despite its promising results, the model also opens several avenues for future exploration to advance fault detection in IoT-enabled infrastructures. Explainability is another crucial aspect that warrants attention. Integrating explainable AI techniques would make the fault detection process more transparent, enabling operators to understand the reasoning behind the model's predictions. This not only fosters trust but also aids in diagnosing root causes more effectively. Predictive maintenance capabilities could also be integrated into the system to transition from reactive to proactive fault management, leveraging time-series data to anticipate and mitigate faults before it occurs. The assessment primarily depends on simulated data, which may not fully reflect real-world complexities such as sensor degradation and communication problems, thus affecting its generalizability. The fault injections adhere to predefined patterns that may not encompass the entire spectrum of industrial fault behaviours. Future research will involve validating the model with actual industrial data, investigating online learning, and evaluating constraints related to practical deployment.

References

1. Wang, Wilson. (2008) An intelligent system for machinery condition monitoring. *IEEE Transactions on Fuzzy Systems* 16, no. 1, 110-122. <https://doi.org/10.1109/TFUZZ.2007.896237>
2. Goyal, Deepam, Anurag Chaudhary, Rajeev Kumar Dang, B. S. Pabla, and S. S. Dhama.(2018). Condition monitoring of rotating machines: a review. *World Scientific News* 113, 98-108.
3. Kumar, Satish, Tushar Kolekar, Shruti Patil, Arunkumar Bongale, Ketan Kotecha, Atef Zaguia, and Chander Prakash. (2022). A low-cost multi-sensor data acquisition system for fault detection in fused deposition modelling. *Sensors* 22, no. 2, 517. <https://doi.org/10.3390/s22020517>
4. Angelopoulos, Angelos, Emmanouel T. Michailidis, Nikolaos Nomikos, Panagiotis Trakadas, Antonis Hatziefremidis, Stamatis Voliotis, and Theodore Zahariadis. (2019). Tackling faults in the industry 4.0 era—a survey of machine-learning solutions and key aspects. *Sensors* 20, no. 1, 109. <https://doi.org/10.3390/s20010109>
5. Cakir, Mustafa, Mehmet Ali Guvenc, and Selcuk Mistikoglu. (2021) The experimental application of popular machine learning algorithms on predictive

- maintenance and the design of IIoT based condition monitoring system. *Computers & Industrial Engineering* 151, 106948. <https://doi.org/10.1016/j.cie.2020.106948>
6. Bahel, Vedant, and Arunesh Mishra. (2021) Ci-mcmts: Computational intelligence based machine condition monitoring system. In 2021 International Conference on Computational Intelligence and Knowledge Economy (ICCIKE), pp. 489-493. IEEE, 2021. <https://doi.org/10.1109/ICCIKE51210.2021.9410744>
 7. Onel, Melis, Chris A. Kieslich, and Efstratios N. Pistikopoulos. (2019). A nonlinear support vector machine-based feature selection approach for fault detection and diagnosis: Application to the Tennessee Eastman process. *AIChE Journal* 65, no. 3, 992-1005. <https://doi.org/10.1002/aic.16497>
 8. Sun, Jian, Teja Kuruganti, Brian Fricke, Shenglan Xuan, Yanfei Li, Wayne Wilkerson, and Carlos Cunningham. (2022) Automated fault detection and diagnosis deployment Internet of Things solution for building energy system. *Journal of Building Engineering* 61, 105291. <https://doi.org/10.1016/j.jobbe.2022.105291>
 9. Almutairi, Khalid, Jyoti K. Sinha, and Haobin Wen. (2024) Fault Detection of Rotating Machines Using poly-Coherent Composite Spectrum of Measured Vibration Responses with Machine Learning. *Machines* 12, no. 8, 573. <https://doi.org/10.3390/machines12080573>
 10. Alharbi, Fahad, Suhuai Luo, Hongyu Zhang, Kamran Shaukat, Guang Yang, Craig A. Wheeler, and Zhiyong Chen. (2023) A brief review of acoustic and vibration signal-based fault detection for belt conveyor idlers using machine learning models. *Sensors* 23, no. 4, 1902. <https://doi.org/10.3390/s23041902>
 11. Saleem, F., Umar, M., & Kim, J. M. (2025). An Optimized Few-Shot Learning Framework for Fault Diagnosis in Milling Machines. *Machines*, 13(11), 1010. <https://doi.org/10.3390/machines13111010>
 12. Saleem, F., & Kim, J. M. (2025, November). Two-Stage Graph Convolutional Framework for Pipeline Leak Detection and Size Identification. In *International Conference on Innovative Techniques and Applications of Artificial Intelligence* (pp. 229-242). Cham: Springer Nature Switzerland. https://doi.org/10.1007/978-3-032-11402-0_17
 13. Neupane, Dhiraj, Mohamed Reda Bouadjenek, Richard Dazeley, and Sunil Aryal. (2024) Data-driven Machinery Fault Detection: A Comprehensive Review." *arXiv preprint arXiv:2405.18843*.
 14. Moshrefi, Amirhossein, and Frederic Nabki. (2024) Advanced industrial fault detection: A comparative analysis of ultrasonic signal processing and ensemble machine learning techniques. *Applied Sciences* 14, no. 15, 6397. <https://doi.org/10.3390/app14156397>
 15. S. Basangar and B. N. Tripathi, (2020) Literature Review on Fault Detection of Equipment using Machine Learning Techniques. 2020 International Conference on Computation, Automation and Knowledge Management (ICCAKM), Dubai, United Arab Emirates, 2020, pp. 62-67. <https://doi.org/10.1109/ICCAKM46823.2020.9051543>
 16. Cui, Wei, Guoying Meng, Aiming Wang, Xinge Zhang, and Jun Ding. (2021) Application of rotating machinery fault diagnosis based on deep learning. *Shock and Vibration* 2021, no. 1, 3083190. <https://doi.org/10.1155/2021/3083190>
 17. Laaradj, Sail Hadj, Lousdad Abdelkader, Bouamama Mohamed, and Nouioua Mourad. (2023) Vibration-based fault diagnosis of dynamic rotating systems for real-time maintenance monitoring. *The International Journal of Advanced Manufacturing Technology* 126, no. 7, 3283-3296. <https://doi.org/10.1007/s00170-023-11320-5>
 18. Sohaib, Muhammad, Shahid Munir, MM Manjurul Islam, Jungpil Shin, Faisal Tariq, SM Mamun Ar Rashid, and Jong-Myon Kim. (2022) Gearbox fault diagnosis

- using improved feature representation and multitask learning. *Frontiers in Energy Research* 10, 998760. <https://doi.org/10.3389/fenrg.2022.998760>
19. Qiu, Shaohua, Xiaopeng Cui, Zuowei Ping, Nanliang Shan, Zhong Li, Xianqiang Bao, and Xinghua Xu. (2023) Deep learning techniques in intelligent fault diagnosis and prognosis for industrial systems: a review. *Sensors* 23, no. 3, 1305. <https://doi.org/10.3390/s23031305>
 20. Moosavi, Sajad, Maryam Farajzadeh-Zanjani, Roozbeh Razavi-Far, Vasile Palade, and Mehrdad Saif. (2024) Explainable AI in Manufacturing and Industrial Cyber-Physical Systems: A Survey. *Electronics* 13, no. 17, 3497. <https://doi.org/10.3390/electronics13173497>
 21. S. K. Perumal and S. Meenakshi (2023) Intelligent Machine Fault Detection in Industries using Supervised Machine Learning Techniques. 2023 International Conference on Data Science, Agents & Artificial Intelligence (ICDSAAI), Chennai, India, 2023, pp. 1-5, <https://doi.org/10.1109/ICDSAAI59313.2023.10452465>
 22. Priya, P.I., Muthurajkumar, S. & Daisy, S.S. (2022) Data Fault Detection in Wireless Sensor Networks Using Machine Learning Techniques. *Wireless Pers Commun* 122, 2441-2462. <https://doi.org/10.1007/s11277-021-09001-1>
 23. Leite, Denis, Aldonso Martins, Jr., Diego Rativa, Joao F. L. De Oliveira, and Alexandre M. A. Maciel. (2022). An Automated Machine Learning Approach for Real-Time Fault Detection and Diagnosis. *Sensors* 22, no. 16: 6138. <https://doi.org/10.3390/s22166138>
 24. E. U. Warriach and K. Tei (2013) Fault Detection in Wireless Sensor Networks: A Machine Learning Approach, 2013 IEEE 16th International Conference on Computational Science and Engineering, Sydney, NSW, Australia, 2013, pp. 758-765, <https://doi.org/10.1109/CSE.2013.116>.
 25. Inyang, Udeme Ibang, Ivan Petrunin, and Ian Jennions. (2023). Diagnosis of Multiple Faults in Rotating Machinery Using Ensemble Learning. *Sensors* 23, no. 2: 1005. <https://doi.org/10.3390/s23021005>
 26. Miao, Yunchu, Zhen Li, and Maoyin Chen. (2024) Time/Frequency Feature-Driven Ensemble Learning for Fault Detection. *Processes* 12, no. 10: 2099. <https://doi.org/10.3390/pr12102099>
 27. J. P. Jose, T. Ananthan and N. K. Prakash (2022) Ensemble Learning Methods for Machine Fault Diagnosis. 2022 Third International Conference on Intelligent Computing Instrumentation and Control Technologies (ICICICT), Kannur, India, 2022, pp. 1127-1134, <https://doi.org/10.1109/ICICICT54557.2022.9917966>
 28. Umar, M., Siddique, M. F., & Kim, J. M. (2025). Burst-informed acoustic emission framework for explainable failure diagnosis in milling machines. *Engineering Failure Analysis*, 110373. <https://doi.org/10.1016/j.engfailanal.2025.110373>
 29. Siddique, M. F., Umar, M., Ahmad, W., & Kim, J. M. (2025). Advanced fault diagnosis in milling cutting tools using vision transformers with semi-supervised learning and uncertainty quantification. *Scientific Reports*, 15(1), 42460. <https://doi.org/10.1038/s41598-025-26550-8>
 30. Siddique, M. F., Zaman, W., Umar, M., Kim, J. Y., & Kim, J. M. (2025). A hybrid deep learning framework for fault diagnosis in milling machines. *Sensors*, 25(18), 5866. <https://doi.org/10.3390/s25185866>
 31. Ahmad, Z., Ullah, N., Zaman, W., Siddique, M. F., Kim, J., & Kim, J. M. (2023, November). A Framework for Centrifugal Pump Diagnosis Using Health Sensitivity Ratio Based Feature Selection and KNN. In *Asian Conference on Pattern Recognition* (pp. 170-179). Cham: Springer Nature Switzerland. https://doi.org/10.1007/978-3-031-47637-2_13
 32. Zhou, Zijun, Qingsong Ai, Ping Lou, Jianmin Hu, and Junwei Yan. (2024) A Novel Method for Rolling Bearing Fault Diagnosis Based on Gramian Angular Field and CNN-ViT. *Sensors* 24, no. 12: 3967. <https://doi.org/10.3390/s24123967>
 33. Tawakuli, Amal, Bastian Havers, Vincenzo Gulisano, Daniel Kaiser, and Thomas Engel (2024) Survey: time-series data preprocessing: a survey and an empirical

- analysis. *Journal of Engineering Research*
<https://doi.org/10.1016/j.jer.2024.02.018>
34. Chen, Xiaoyang, Meng Wang, and Huitao Zhang (2024) Machine Learning-based Fault Prediction and Diagnosis of Brushless Motors. *Engineering Advances* 4, no. 3, 130-142.
 35. Rajak, Anjali, and Rakesh Tripathi. (2024) FDF-HybridFS: Towards design of a failure detection framework using hybrid feature selection method for IP core networks that connect 5G core in NFV-based test environment. *Computer Standards & Interfaces* 87, 103779. <https://doi.org/10.1016/j.csi.2023.103779>
 36. Zhao, Huimin, Jianjie Zheng, Junjie Xu, and Wu Deng (2019) Fault diagnosis method based on principal component analysis and broad learning system. *IEEE Access* 7, 99263-99272. <https://doi.org/10.1109/ACCESS.2019.2929094>
 37. Jha, Vidyapati, and Priyanka Tripathi. (2024) Decentralized knowledge discovery using massive heterogenous data in Cognitive IoT. *Cluster Computing* 27, no. 3, 3657-3682. <https://doi.org/10.1007/s10586-023-04154-z>
 38. Luo, Lijia, Weida Wang, Shiyi Bao, Xin Peng, and Yigong Peng. (2024) Robust and sparse canonical correlation analysis for fault detection and diagnosis using training data with outliers. *Expert Systems with Applications* 236, 121434. <https://doi.org/10.1016/j.eswa.2023.121434>
 39. Genel, Kübra Bağcı, and H. Eray Çelik. (2024) An Application of Robust Principal Component Analysis Methods for Anomaly Detection. *Turkish Journal of Science and Technology* 19, no. 1, 107-112. <https://doi.org/10.55525/tjst.1293057>
 40. Geetha, G., and P. Geethanjali. (2024) An efficient method for bearing fault diagnosis. *Systems Science & Control Engineering* 12, no. 1, 2329264. <https://doi.org/10.1080/21642583.2024.2329264>
 41. Jing, Chen, and Jian Hou. (2015) SVM and PCA based fault classification approaches for complicated industrial process. *Neurocomputing* 167, 636-642. <https://doi.org/10.1016/j.neucom.2015.03.082>
 42. Martin-Diaz, Ignacio, Daniel Morinigo-Sotelo, Oscar Duque-Perez, and Rene de J. Romero-Troncoso. (2016) Early fault detection in induction motors using AdaBoost with imbalanced small data and optimized sampling. *IEEE Transactions on Industry Applications* 53, no. 3, 3066-3075. <https://doi.org/10.1109/TIA.2016.2618756>
 43. Long, Zhuo, Xiaofei Zhang, Li Zhang, Guojun Qin, Shoudao Huang, Dianyi Song, Haidong Shao, and Gongping Wu. (2021) Motor fault diagnosis using attention mechanism and improved adaboost driven by multi-sensor information. *Measurement* 170, 108718. <https://doi.org/10.1016/j.measurement.2020.108718>
 44. Mittal, Khushi, Kanwarpartap Singh Gill, Rahul Chauhan, and Hemant Singh Pokhariya. (2024) Exploring AdaBoost's Classifier's Power in Electrical Fault Detection: A Comprehensive Analysis. In *2024 International Conference on Innovations and Challenges in Emerging Technologies (ICICET)*, pp. 1-5. IEEE, 2024. <https://doi.org/10.1109/ICICET59348.2024.10616282>
 45. Sivagamasundari, S., and Manjula Sri Rayudu. (2023) IoT based solar panel fault and maintenance detection using decision tree with light gradient boosting. *Measurement: Sensors* 27, 100726. <https://doi.org/10.1016/j.measen.2023.100726>
 46. Yaqub, Raziq, Hassan Ali, and Mohd Helmy Bin Abd Wahab. (2023) Electrical Motor Fault Detection System using AI's Random Forest Classifier Technique. In *2023 IEEE International Conference on Advanced Systems and Emergent Technologies (IC_ASET)*, pp. 1-5. IEEE, 2023. https://doi.org/10.1109/IC_ASET58101.2023.10150924
 47. Al Tobi, M. A. S., K. P. Ramachandran, S. Al-Araimi, Rene Pacturan, Amuthakannan Rajakannu, and Chetha Achuthan. (2022) Machinery faults diagnosis using support vector machine (SVM) and Naïve Bayes classifiers. *Int. J. Engi. Trends Technol.* 70, no. 12, 26-34. <https://doi.org/10.14445/22315381/IJETT-V70I12P204>

48. Mujib, Achmad, and Taufik Djatna. (2020) Ensemble learning for predictive maintenance on wafer stick machine using IoT sensor data. In 2020 International Conference on Computer Science and Its Application in Agriculture (ICOSICA), pp. 1-5. IEEE, 2020. <https://doi.org/10.1109/ICOSICA49951.2020.9243180>
49. Moradi, A., Tajalli, S., Mosallanejad, M. H., & Saboori, A. (2024). Intelligent laser-based metal additive manufacturing: A review on machine learning for process optimization and property prediction. *The International Journal of Advanced Manufacturing Technology*, 1-34. <https://doi.org/10.1007/s00170-024-14858-0>
50. Saifan, A., Chen, S., Miao, H., Shen, Y., Yuan, L., Zhu, R., & Yang, G. (2024). Data-driven modeling and optimization of a robotized multi-needle ultrasonic peen-forming process for 2024-T3 aluminum alloy. *The International Journal of Advanced Manufacturing Technology*, 1-23. <https://doi.org/10.1007/s00170-024-14438-2>
51. Chang, L. K., Chen, R. S., Tsai, M. C., Lee, R. M., Lin, C. C., Huang, J. C., ... & Horng, M. H. (2024). Machine learning applied to property prediction of metal additive manufacturing products with textural features extraction. *The International Journal of Advanced Manufacturing Technology*, 132(1), 83-98. <https://doi.org/10.1007/s00170-024-13165-y>

Author contributions

Devi Gayadri, Kanagaraj, Jayant Giri : Writing original draft , review , editing , methodology , investigation Mohammad Kanan : Writing review , editing , formal analysis , funding , data analysis.

Data Availability

The datasets generated during and /or analysed during the current study are available from the corresponding author upon reasonable request.

Competing interests

The authors declare no competing interests.

Ethical approval and consent to participate

This article does not contain any studies with human participants or animals performed by any of the authors. All the authors have read and approved the final manuscript.

Additional information

Correspondence and requests for materials should be addressed to G.K. or M.K

ARTICLE IN PRESS