



## Machine fault detection through sound analysis using MFCC and machine learning

Steven Henderson Chang<sup>1</sup>, Ariana Tulus Purnomo<sup>2</sup>, Muhammad Agni Catur Bhakti<sup>2</sup>, Vania Katherine Mulia<sup>3</sup>, Agyl Fajar Rizky<sup>4</sup>, Nikolas Krisma Hadi Fernandez<sup>1</sup>, Farid Triawan<sup>1,\*</sup>

<sup>1</sup>Department of Mechanical Engineering, University of Sampoerna, Jakarta 12780, Indonesia

<sup>2</sup>Department of Computer Science, University of Sampoerna, Jakarta 12780, Indonesia

<sup>3</sup>Polytechnic School, University of Girona, Catalonia 17004, Spain

<sup>4</sup>PT. Denso Manufacturing Indonesia, Bekasi, West Java, Indonesia

\*Corresponding author: [farid.triawan@sampoernauniversity.ac.id](mailto:farid.triawan@sampoernauniversity.ac.id)

### Abstract

This study addresses the need for automated damage and failure detection in industrial machinery through sound analysis and machine learning. Traditional methods rely on human experts to identify faults using microphones, which can be time-consuming, stressful, and prone to errors such as limited perception, subjectivity, and inconsistency. This study leverages machine learning to create a more objective and efficient alternative. Mel-Frequency Cepstral Coefficients (MFCCs) were employed for feature extraction, capturing intricate sound patterns associated with machinery faults. Through rigorous experimentation, 11 MFCC coefficients were identified as optimal. The Support Vector Machine (SVM) emerged as the best-performing classifier compared to LightGBM and XGBoost, achieving a training accuracy of 83.12% and testing accuracy of 82.50%. The dataset was split between 80% for training and 20% for testing. The small gap between training and testing accuracy indicates an ideal model with no signs of over fitting, under fitting, or data leakage. Real-world simulations validated the model's efficacy under various operational scenarios, demonstrating its readiness for industrial deployment. This study highlights the effectiveness of sound analysis and SVM classification in proactive maintenance, offering a reliable tool to reduce downtime and maintenance costs while enhancing operational efficiency and reliability.

### Keywords:

Fault diagnosis, machine learning, Mel-Frequency Cepstral Coefficients, sound analysis, Support Vector Machine

## 1 Introduction

Damage and failure detection in industrial machinery has become increasingly critical in modern manufacturing environments, as it directly impacts operational efficiency, workplace safety, and productivity [1]. The industrial landscape has undergone significant transformation, aligning with the ninth Sustainable Development Goal-Industry, Innovation, and Infrastructure [2]. Within this context, the field of industrial maintenance has seen remarkable advancements through the integration of cutting-edge technologies [3]. Traditional maintenance methods, relying on scheduled inspections and manual diagnosis, have proven inadequate for modern industrial needs, being time-consuming, error-prone, and limited by human perception and subjectivity [4]. These approaches often depend on human experts to recognize damage through noise

analysis, which is stress-inducing and susceptible to human error due to limited perception, subjectivity, inconsistency, and inefficiency. The limitations of traditional approaches are further exacerbated by the increasing complexity of industrial systems and the growing demand for uninterrupted operational efficiency [5].

In response to these challenges, researchers have turned to non-invasive, data-driven approaches, leveraging advancements in sound analysis and machine learning [6], [7]. Sound analysis, or acoustic monitoring, involves capturing and analyzing machinery sounds during operation, providing crucial information about equipment performance and condition [8], [9]. By examining the frequency, amplitude, and other characteristics of these sounds, it becomes possible to identify operational deviations from normal operation and pinpoint potential defects, fundamentally changing maintenance and defect detection practices [10], [11].

Machine learning techniques enhance sound analysis by handling large datasets and discerning patterns that might elude human observers [12]. These models can be trained to recognize specific sound patterns indicative of machinery faults or inefficiencies [13], [14]. Once trained, they can operate autonomously, continuously monitoring equipment and alerting operators to anomalies [15].

However, challenges persist in fully realizing the potential of this approach. These include determining optimal machine learning models for classifying time series audio data, selecting relevant features from audio signals, and developing comprehensive systems capable of handling diverse acoustic signatures under various operating conditions [16], [17].

This research aims to address these challenges by developing and analyzing a machine learning-based system for automated damage and failure detection through sound analysis in industrial machinery. The study will explore advanced feature extraction techniques for classifying audio signals with similar frequencies and evaluate the performance of machine-learning algorithms suitable for handling time-series audio data. By tackling these complex issues, this research seeks to contribute to the advancement of industrial maintenance practices, enhancing efficiency, reliability, and safety in industrial operations.

## 2 Literature review

This chapter examines machine fault diagnosis through sound analysis, a critical component in damage and failure detection. Sound analysis has become essential in predictive maintenance, allowing early detection of machinery abnormalities [18], [19]. This non-invasive technique captures and analyzes acoustic emissions during machine operation. The process involves sound capture, digitization, feature extraction, and analysis using signal processing and machine learning techniques [20], [21].

Sound analysis's effectiveness stems from its ability to detect early-stage faults, crucial for preemptive maintenance and minimizing downtime costs [6], [9], [22]. Recent advancements in digital signal processing and machine learning have significantly improved its accuracy and efficiency [20]. These algorithms enable systems to recognize and classify acoustic patterns, identifying faults across various industrial settings [19], [23], [24]. This study utilizes these advanced techniques, implemented through Python programming, to develop a robust system for industrial machinery damage and failure detection.

### 2.1 Mel Frequency Cepstral Coefficient

Mel Frequency Cepstral Coefficients (MFCCs) are a widely used method for feature extraction in audio and speech processing [25]. MFCCs represent the short-term power spectrum of sound through a linear cosine transform of a log power spectrum on a nonlinear mel scale of frequency, effectively capturing the spectral envelope [25].

The process of deriving MFCCs involves a series of steps, as illustrated in Fig. 1. First, the audio signal undergoes Fourier transformation, converting it from the time domain to the frequency domain. The powers of the resulting spectrum are then mapped onto the mel scale using triangular overlapping windows. Subsequently,

the logarithms of the powers at each mel frequency are taken, followed by the discrete cosine transform of the list of mel log powers [26], [27]. The resulting amplitudes constitute the MFCCs, providing a compact yet informative representation of the audio signal's spectral characteristics.

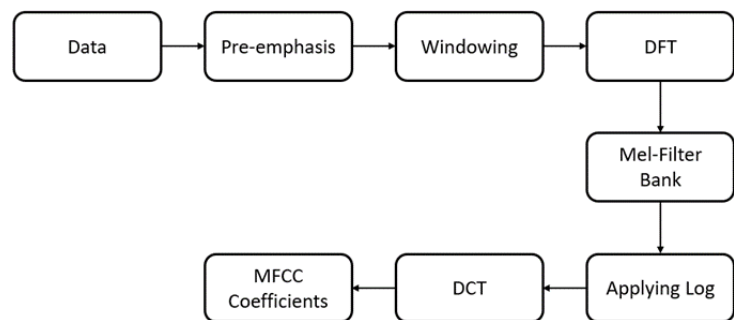


Fig. 1. MFCC framework illustration.

MFCCs are preferred due to their ability to mimic the human auditory system's response, offering a more intuitive representation of the spectral envelope compared to techniques like the Fast Fourier Transform (FFT) [28]. This makes them particularly useful in speech recognition and music information retrieval [28]. In the context of damage and failure detection through sound analysis, integrating MFCCs with machine learning and Python programming is expected to enhance fault detection accuracy and efficiency in industrial settings [26], [29].

## 2.2 Support Vector Machine

Support Vector Machine (SVM) is a prominent-supervised learning algorithm used in machine fault diagnosis through sound analysis [24], [30]. It handles linear and nonlinear classification, regression, and outlier detection tasks [8]. SVM's core principle involves identifying an optimal hyperplane in a high-dimensional feature space to separate data classes, maximizing margins between them [24]. This approach enables SVM to capture complex relationships in sound data, enhancing its ability to generalize across diverse fault patterns [24], [31]. SVM's adaptability to multi-class classification is advantageous for identifying various fault types in industrial machinery [31], [32], while its interpretability allows engineers to gain insights into critical acoustic features contributing to fault identification [24].

## 2.3 Light Gradient Boosting Machine

Light Gradient Boosting Machine (LightGBM) is an advanced, efficient machine learning algorithm gaining popularity in machine fault diagnosis through sound analysis [30]. Developed by Microsoft, LightGBM is renowned for its speed and scalability, making it suitable for large datasets and high-dimensional feature spaces [33]. It employs gradient boosting, building an ensemble of weak learners sequentially to address previous errors [33], [34], [35]. LightGBM's innovative techniques, such as histogram-based learning and leaf-wise tree growth, contribute to its computational efficiency and ability to manage large-scale datasets [15], [33]. The integration of LightGBM into the project framework strategically enhances the accuracy and real-time processing capabilities of the damage and failure detection system.

## 2.4 EXtreme Gradient Boosting

EXtreme Gradient Boosting (XGBoost) is a highly effective machine learning algorithm widely used in machine fault diagnosis through sound analysis [30]. Developed by Tianqi Chen in 2016, XGBoost is renowned for its performance, versatility, and ability to handle various machine learning tasks with high accuracy [36]. Its strength lies in regularization techniques that prevent over fitting and improve generalization by incorporating penalty terms into the objective function [37]. In machine fault diagnosis, XGBoost excels at identifying subtle sound cues indicative of potential faults in industrial machinery, using MFCCs extracted via Python

programming [8]. Incorporating XGBoost into the project framework aligns with the objectives, highlighting its importance in advancing predictive maintenance and enhancing industrial machinery resilience.

## 2.5 Related works

The MFCCs were selected for feature extraction due to their proven effectiveness in capturing spectral envelopes of sound signals, making them suitable for industrial machinery fault detection [25], [28]. The MFCCs extraction process aligns with the project's goal of enhancing fault detection accuracy [26], [27].

SVM was chosen for its robust performance in high-dimensional spaces and ability to handle non-linear relationships. SVM's optimal hyper plane identification enables it to capture nuanced acoustic characteristics indicative of potential faults [8], [24]. Its versatility in multi-class classification suits the identification of various fault types [31], [32].

LightGBM was selected for its speed and efficiency in handling large-scale data. LightGBM's histogram-based learning and leaf-wise tree growth techniques offer superior computational efficiency [33]. Its Gradient-based One-Side Sampling (GOSS) and Exclusive Feature Bundling (EFB) address limitations in traditional Gradient-Boosting Decision Tree frameworks [35].

XGBoost was included based on its proven performance in machine learning competitions and ability to handle complex data relationships. XGBoost enhances weak learners through iterative improvement and incorporates regularization techniques to prevent overfitting [36], [37]. Its depth-first tree construction approach and sparsity-aware split finding technique enable it to capture intricate patterns in sound data efficiently.

The integration of these methods within Python enhances the project's real-world applicability, leveraging extensive libraries for scientific computing, data analysis, and machine learning to facilitate efficient sound processing and fault diagnosis [30], [38]. This approach, combining advanced feature extraction with state-of-the-art machine learning, advances predictive maintenance strategies. By leveraging Python's capabilities to process and analyze sounds efficiently, the study aims to develop a robust system for damage and failure detection in industrial machinery, contributing to improved operational resilience and efficiency in industrial contexts.

## 3 Research methodology

This chapter outlines the systematic methodology employed to leverage sound analysis for detecting damage and failure in industrial machinery, specifically a DC motor. As depicted in Fig. 2, the author's approach encompasses various stages, from data gathering and database creation to feature extraction, data splitting, model training, evaluation, and refinement.

### 3.1 Data gathering

The foundational stage of the methodological framework centered on the meticulous acquisition of acoustic data from diverse sources. This critical process, termed Data Gathering encompasses the collection of sound signals emanating from DC motors operating under varying conditions. Sound signals from a DC motor were recorded under different conditions at a Japanese automotive company. The author categorized machine sounds into two classes: okay or good (OK) and not good (NG), each with two subclasses: clockwise (CW) and non-clockwise (CCW).

The experimental setup, as shown in Fig. 3, utilized a laptop as the data acquisition card, capturing signals at 48 kHz to cover the audible range. Positioned 10 cm from the sound source, the laptop utilized a microphone cable for recording. A total of 400 sound recordings were meticulously collected from the DC motor under various operational conditions, as detailed in Table 1. This systematic categorization ensures a diverse and comprehensive representation of the machine's sound profile, with 200 recordings in each of the four subclasses (CW\_OK, CW\_NG, CCW\_OK, and CCW\_NG).

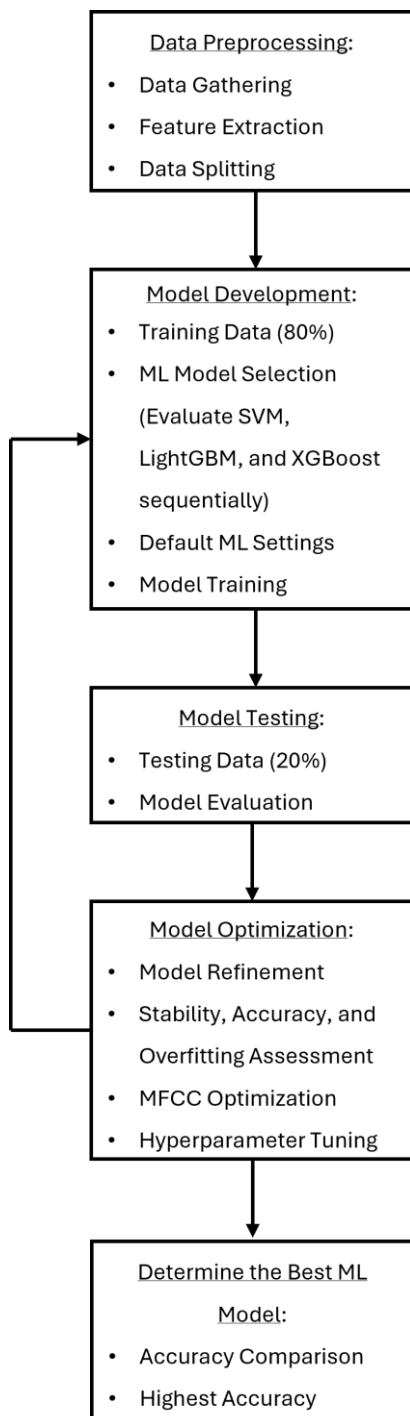


Fig. 2. Block diagram of the research methodology.

Table 1. Distribution of sound recordings

Class	CW	CCW	Total
OK	100	100	200
NG	100	100	200
Total	200	200	400

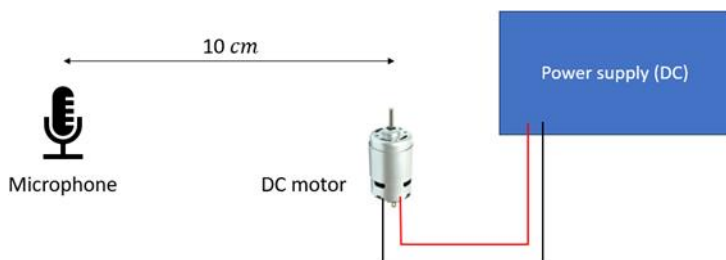


Fig. 3. Experimental setup.

### 3.2 Feature extraction

The second stage of the methodological framework involves extracting MFCCs from the recorded sound signals in the dataset. Leveraging Python programming, the feature extraction process

employs a series of signal processing steps to transform the raw audio recordings into meaningful features suitable for machine learning analysis.

The process begins with pre-emphasis, applying a high-pass filter to enhance high-frequency components in the audio signal. The audio data then undergoes windowing, dividing it into overlapping frames to facilitate the analysis of short-time features. Next, a Discrete Fourier Transform (DFT) is applied to each frame via the FFT, resulting in a spectrogram that captures the signal's frequency content over time. By using Mel-filter banks, which mimic human ear sensitivity, the process captures the energy distribution across frequency bands, aligning with human auditory perception for compact spectral representation. Applying a logarithmic transformation enhances the feature's discriminative power by compressing the data range and emphasizing spectral variations. Finally, a Discrete Cosine Transform (DCT) is employed to extract 13 MFCCs, encapsulating the essential spectral features of the audio waveform [28], [39].

These MFCCs serve as the transformed representation of the original sound recordings, containing the essential characteristics and patterns derived from the audio signals. The dataset, now with the extracted MFCC features, is set and readied for integration into various machine learning algorithms for further analysis.

### 3.3 Data splitting

Following the feature extraction stage, the DC motor sound recording's dataset is systematically partitioned into distinct subsets for model training and evaluation. This careful partitioning ensures data integrity and supports effective feature extraction and analysis for the fault detection and diagnosis system.

To ensure robust model training and evaluation, the dataset is split into 80% for training and 20% for testing, as shown in Table 2. This allows the model to learn patterns from most of the data while using an independent set to evaluate generalization. The labels 'CW\_OK,' 'CW\_NG,' 'CCW\_OK,' and 'CCW\_NG' for the DC motor sound recordings are grouped into broader categories: 'OK' (for CW\_OK and CCW\_OK) and 'NG' (for CW\_NG and CCW\_NG). This transformation aids the model in accurately classifying the DC motor's operational conditions.

Table 2. Training and testing data allocation

Set	Class	CW	CCW	Total
Training	OK	80	80	160
	NG	80	80	160
Testing	OK	20	20	40
	NG	20	20	40
Total		200	200	400

### 3.4 Model train

In the fourth stage, the author trains a machine learning model using 80% of the collected sound recordings from the DC motor as the training data set. The training process involves extracting MFCC features from the sound recordings, employing the machine-learning algorithms, and using cross-validation to optimize the models' performance and ensure its robustness in discerning patterns and classifying the operational conditions of the DC motors, setting the stage for subsequent stages using the reserved testing dataset.

### 3.5 Model evaluation

In the fifth stage, the author thoroughly evaluates the performance of the trained machine learning model. The evaluation process involves ingesting the reserved 20% testing dataset and using the extracted MFCC features as inputs to the pre-trained model. Key evaluation metrics are employed, including the confusion matrix, which provides a visual representation of the model's predictive accuracy across different classes, and the overall model accuracy calculated by comparing predicted and actual test labels. In the matrix, label 0 represents "Not OK" (NG) status and

label 1 indicates "OK" status, showing how well the model distinguishes between satisfactory and unsatisfactory conditions.

This detailed assessment enables a deeper understanding of the model's predictive prowess across diverse operational conditions of the machine system. The combination of visual aids and numerical accuracy scores offers valuable insights into the model's classification capabilities.

### 3.6 Model refinement

In the final stage, the author refines the machine learning model through MFCC optimization and hyper parameter tuning to maximize the model's accuracy in classifying sound signals from DC motors. The goal is to achieve a balance between sensitivity and specificity, address over fitting, under fitting, or data leakage issues, and improve the model's generalization, ensuring its reliability in real applications and meeting the research objectives.

### 3.7 MFCC optimization

In the model refinement process, the author focuses on optimizing the number of MFCCs to determine the ideal quantity for accurate fault detection in the motor sound signals. MFCCs are widely used in audio processing tasks, as they represent the short-term power spectrum of a sound based on a linear cosine transform of a log power spectrum on a nonlinear mel scale of frequency. The author will test a range of 7 to 20 MFCCs, where the lower-order coefficients capture the overall spectral shape or envelope, while the higher-order coefficients represent finer spectral details [29]. By systematically evaluating the impact of different MFCC quantities on model accuracy, the author aims to refine the feature extraction process and enhance the model's predictive capabilities for the dataset.

### 3.8 Hyper parameter tuning

In the pivotal model refinement stage, the author engages in meticulous hyper parameter tuning using manual intervention. This hands-on approach leverages the author's domain knowledge and intuition to fine-tune the model's settings, ensuring they are tailored to the specific characteristics of the data and the nuances of the sound signals [40], [41]. The author identifies and prioritizes key hyper parameters, such as learning rates, regularization strengths, and tree depths, which significantly influence the model's learning process and predictive capabilities. By meticulously selecting and adjusting these parameters, the author aims to craft a finely tuned model that strikes an optimal balance between complexity and generalization ability, ensuring robust performance in real-world scenarios.

For each machine learning model, the author prioritizes three key hyper parameters essential for optimizing performance. For SVM, the emphasis is on the kernel type, regularization parameter (C), and kernel coefficient (gamma) [40]. For LightGBM, the author focuses on num\_leaves, learning rate, and n\_estimators, which impact the model's complexity, learning step size, and the number of boosting iterations [41]. Similarly, for XGBoost, the learning rate, max\_depth, and n\_estimators are identified as crucial hyperparameters, affecting the step size, model capacity, and the number of trees. By systematically adjusting these parameters, the author aims to refine one of the models for accurate classification of sound signals from DC motors.

### 3.9 Machine learning model selection

In the final stage of the research, the author methodically evaluated three machine learning models-SVM, LightGBM, and XGBoost-to determine the best model for detecting machine damage and faults. The evaluation process involves scrutinizing the performance of each model across various metrics, as outlined in the previous section, including accuracy, precision, recall, and F1-score.

The primary criterion for selecting the best model is the highest testing accuracy, as this is a fundamental requirement for effective fault detection. Additionally, the author will consider models with a lesser chance of over fitting, under fitting, or data leakage, as these factors can compromise the model's robustness and reliability in real-world application scenarios.

Through this meticulous process of experimentation and evaluation, the author aims to make an informed decision, selecting a model that not only demonstrates superior predictive capabilities within the dataset but also promises to be a reliable and robust solution for real-world applications. By carefully considering the various performance metrics and potential risks, the author will choose the model that best meets the research objectives.

### 3.10 Application

In this stage of the research, the author conducts a real simulation to evaluate the model's performance in classifying new data against the entire dataset. The testing dataset, representing 10% of the dataset, is divided into two scenarios. In Scenario 1, 75% of the data is labeled as 'OK' and 25% as 'NG', while in Scenario 2, this distribution is reversed. These scenarios replicate practical situations where the model encounters additional data and must accurately classify it based on prior training.

Using a setup where 90% of the data serves as the training dataset (considered as the entire dataset in this simulation), and the remaining 10% as the testing dataset (treated as new data), the author aims to assess how effectively the optimized MFCCs and selected optimized machine learning model generalize to new data and distinguish between 'OK' and 'NG' conditions. This evaluation validates the model's robustness and reliability in real applications, ensuring it can handle diverse distributions of unseen data.

## 4 Result and discussion

This chapter presents the research results, including findings from feature extraction, optimization, hyper parameter tuning, model selection, and real application. Results are illustrated through figures, graphs, and tables to clarify the outcomes and their implications for sound-based fault detection in industrial machinery.

### 4.1 Feature extraction

In this section, MFCCs are extracted from DC motor sound recordings. An example of this process will be demonstrated using a file from the CW\_OK class. The original audio waveform for CW\_OK is depicted in Fig. 4, showing a minor delay of 0 to 0.1 seconds due to the recording equipment, which is considered acceptable for this study. The MFCC extraction process involves several steps, including pre-emphasis, framing, windowing, FFT, Mel-filter bank processing, logarithm of the power spectrum, and DCT, with each step detailed in Fig. 5 to Fig. 10.

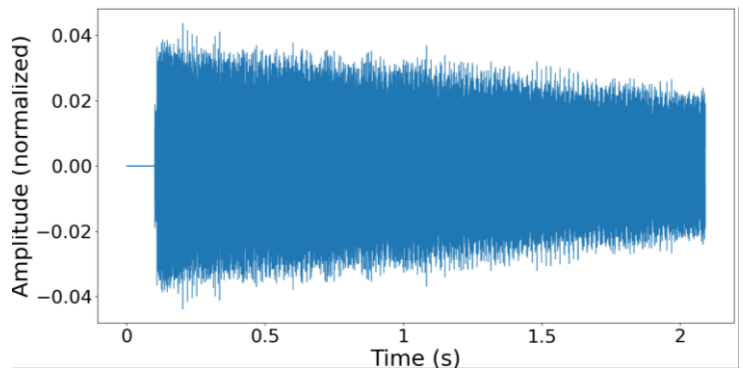


Fig. 4. Original audio graph.

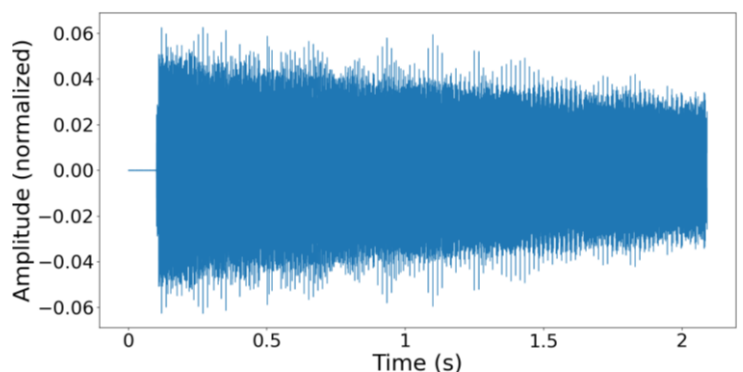


Fig. 5. Pre-emphasized graph.

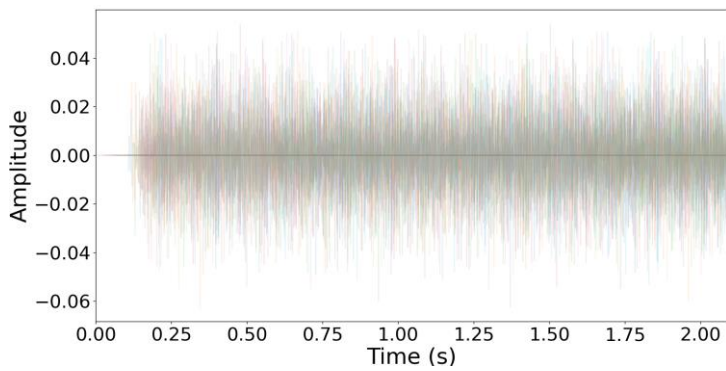


Fig. 6. Overlapping frames graph.

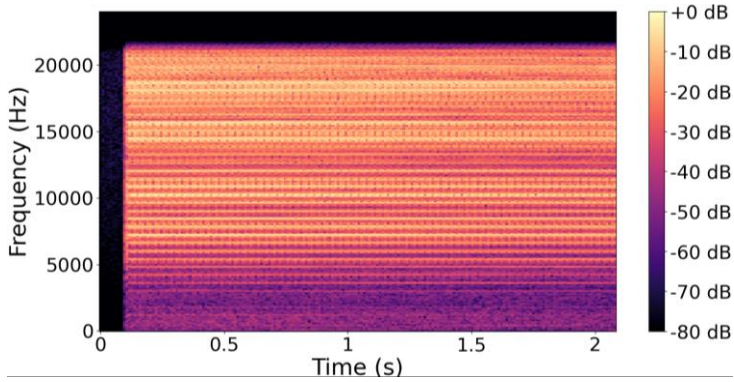


Fig. 7. Spectrogram graph.

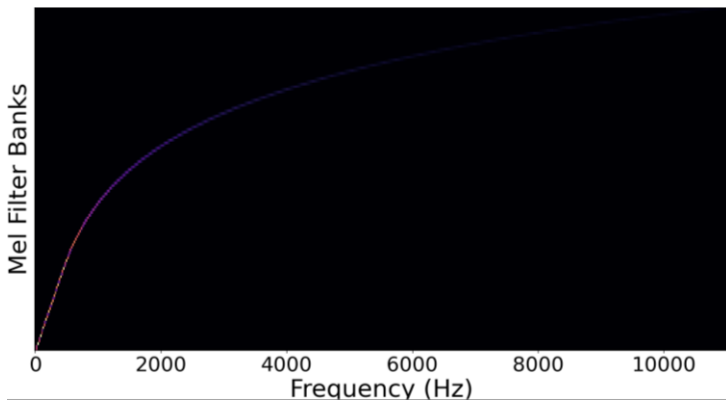


Fig. 8. Mel-filter bank graph

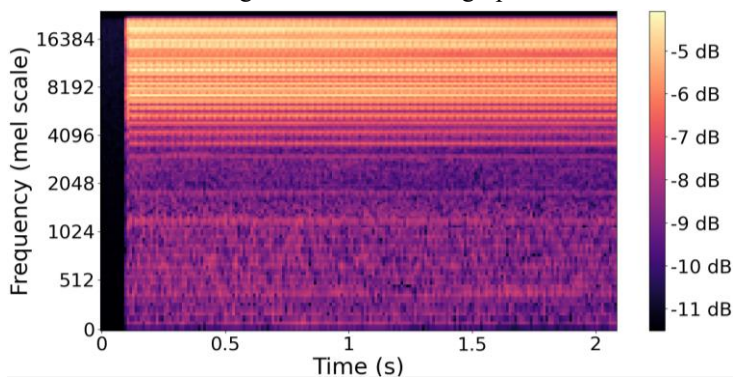


Fig. 9. Log-mel spectrum graph.

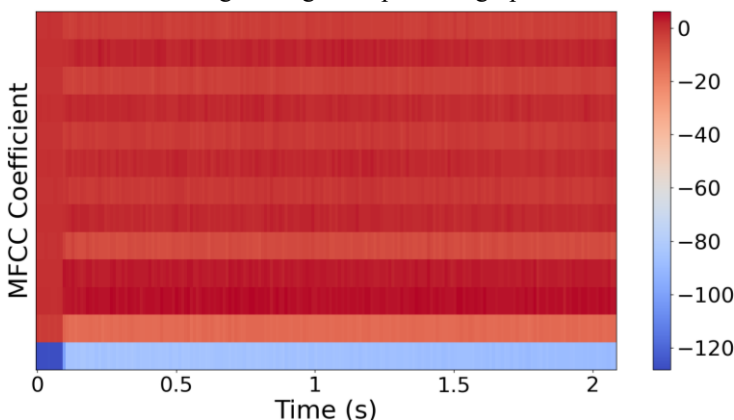


Fig. 10. MFCC graph.

The extracted MFCCs capture distinctive audio characteristics, enabling effective classification of DC motor conditions. This sets the stage for model development. The dataset is split into two, of which 80% are for training and 20% are for testing, ensuring the machine learning models are evaluated on unseen data, providing a realistic assessment of their performance.

## 4.2 Model refinement

Following the feature extraction process, this section delves into the model refinement phase for the dataset. The refinement process includes two key steps: MFCC optimization and hyper parameter tuning. These steps aim to enhance the model's performance and accuracy in detecting faults in industrial machinery

### 4.2.1 MFCC optimization

In the initial step of model refinement, the author focuses on optimizing the number of MFCCs used to capture the distinctive characteristics of the audio signals. Through a series of experiments using SVM, the author tested a range of 7 to 20 MFCC coefficients, including the commonly used default of 13, and evaluated the performance using training and testing accuracies. Table 3 presents the results of the MFCC optimization experiments.

Table 3. MFCC optimization

Coefficient	Train accuracy (%)	Test accuracy (%)
7	76.96	81.25
8	79.06	76.25
9	79.69	80.00
10	78.12	78.75
11	80.94	81.25
12	86.56	77.50
13	85.94	80.00
14	87.19	80.00
15	87.19	78.75
16	88.44	72.50
17	90.00	73.75
18	89.69	73.75
19	90.31	73.75
20	91.56	68.75

The goal is to identify the optimal number of coefficients that maximize the model's accuracy while avoiding over fitting, under fitting, or data leakage. The results show that using 11 MFCC coefficients provides a balanced and high performance, with both training and testing accuracy at around 80%. In contrast, higher numbers of coefficients, such as 12 and above, tend to exhibit higher training accuracy but lower testing accuracies, indicating potential over fitting. On the other hand, lower coefficients, such as 7 and 8, show reasonable testing accuracy but lower training accuracies, which could suggest data leakage. The choice of 11 MFCC coefficients strikes an optimal balance, ensuring the model has enough detail to accurately classify the sound signals without over fitting, and this finding can be adapted for other machine learning models like XGBoost and LightGBM.

### 4.2.2 Hyper parameter tuning

Following the optimization of MFCC, the author moves to the critical step of hyper parameter tuning to fine-tune the machine learning models for optimal performance in classifying sound signals from the dataset. Each model underwent a comprehensive search across a specified hyper parameter space, detailed in Table 4, to determine the best configuration based on training and testing accuracy. The hyper parameter tuning process explored a total of 105 combinations for SVM and 27 combinations each for LightGBM and XGBoost.

In the hyper parameter tuning of the SVM model, the optimal configuration was found to be  $C = 10$ ,  $\gamma = 0.1$ , and kernel = 'rbf', resulting in a testing accuracy of 82.50% and a training accuracy of 83.12%, as shown in Table 5.

Table 4. Hyperparameter search space for each machine learning model

Model	Hyper parameter	Options
SVM	C	0.01, 0.1, 1, 10, 100
	Gamma	'auto', 'scale', 0.001, 0.01, 0.1, 1, 10
	Kernel	'rbf', 'linear', 'sigmoid'
LightGBM	num_leaves	5, 20, 31
	learning_rate	0.05, 0.1, 0.2
	n_estimators	50, 100, 150
XGBoost	learning_rate	0.01, 0.1, 0.3
	max_depth	4, 6, 10
	n_estimators	5, 100, 150

Table 5. SVM accuracy before and after optimization

Parameter	Scenario	
	Default	Optimized
C	1.0	10
Gamma	'scale'	0.1
Kernel	'rbf'	'rbf'
Training accuracy (%)	80.94	83.12
Testing accuracy (%)	81.25	82.50

This setup demonstrated a minimal gap between training and testing accuracies, indicating a balanced model capable of generalizing well to new data. The chosen parameters optimized the model's performance, achieving higher testing accuracy compared to the default configuration (testing accuracy of 81.25% and training accuracy of 80.94%). The confusion matrix in Fig. 11(a) illustrates the model's ability to classify instances into 'OK' and 'NG' categories, with 28 out of 40 'NG' instances correctly classified and 38 out of 40 'OK' instances correctly classified.

For the LightGBM model, the optimal configuration involved setting num\_leaves = 5, learning\_rate = 0.2, and n\_estimators = 50, resulting in a testing accuracy of 81.25% and a training accuracy of 90.00%, as detailed in Table 6. This setup aimed to strike a balance between model complexity and accuracy, enhancing the model's ability to classify sound signals effectively. The decision to adopt these parameters was based on their ability to improve testing accuracy significantly compared to the default setup (testing accuracy of 72.50% and training accuracy of 88.75%), while also reducing the risk of over fitting. The confusion matrix in Fig. 11(b) shows that 30 out of 40 'NG' instances were correctly classified, and 35 out of 40 'OK' instances were correctly classified.

Table 6. LightGBM accuracy before and after optimization

Parameter	Scenario	
	Default	Optimized
num_leaves	31	5
learning_rate	0.1	0.2
n_estimators	100	50
Training accuracy (%)	88.75	90.00
Testing accuracy (%)	72.50	81.25

The XGBoost model's optimal configuration was achieved with learning\_rate = 0.1, max\_depth = 4, and n\_estimators = 100, resulting in a testing accuracy of 76.25% and a training accuracy of 88.75%, as presented in Table 7. This configuration aimed to enhance the model's generalization ability while maintaining high accuracy during training. The parameters were selected based on their ability to improve testing accuracy compared to the default settings (testing accuracy of 75.00% and training accuracy of 89.06%), thereby minimizing the risk of over fitting. The confusion matrix in Fig. 11(c) indicates that 27 out of 40 'NG' instances were correctly classified, and 36 out of 40 'OK' instances were correctly classified.

Table 7. XGBoost accuracy before and after optimization

Parameter	Scenario	
	Default	Optimized
learning_rate	0.03	0.1
max_depth	6	4
n_estimators	100	100
Training accuracy (%)	89.06	88.75
Testing accuracy (%)	75.00	76.25

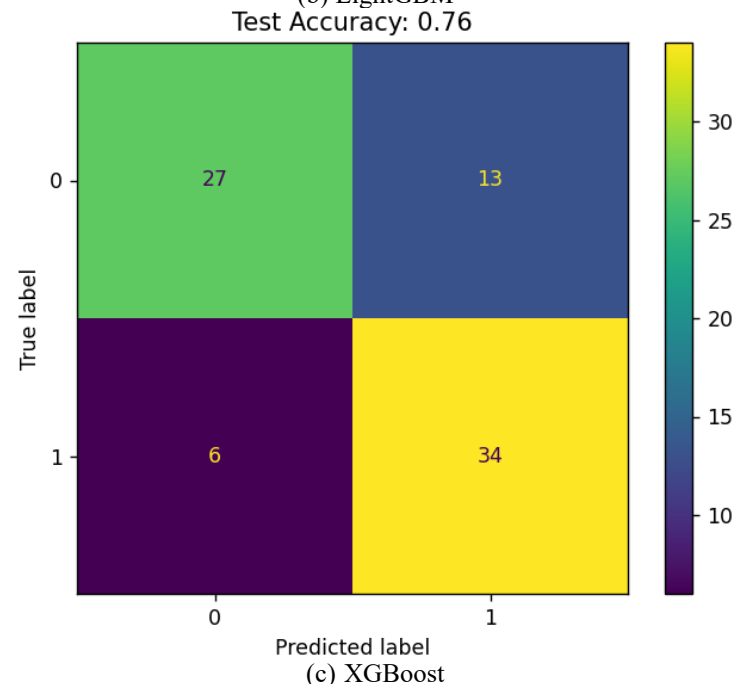
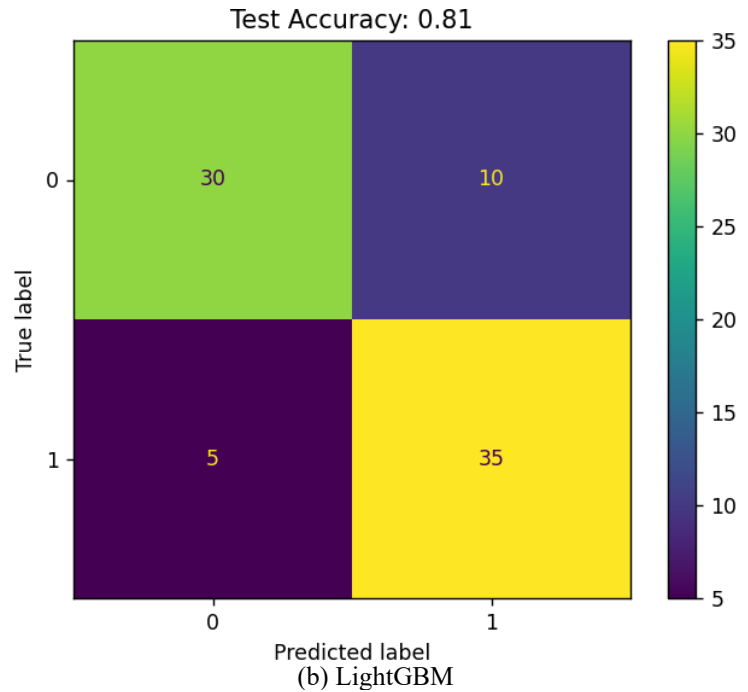
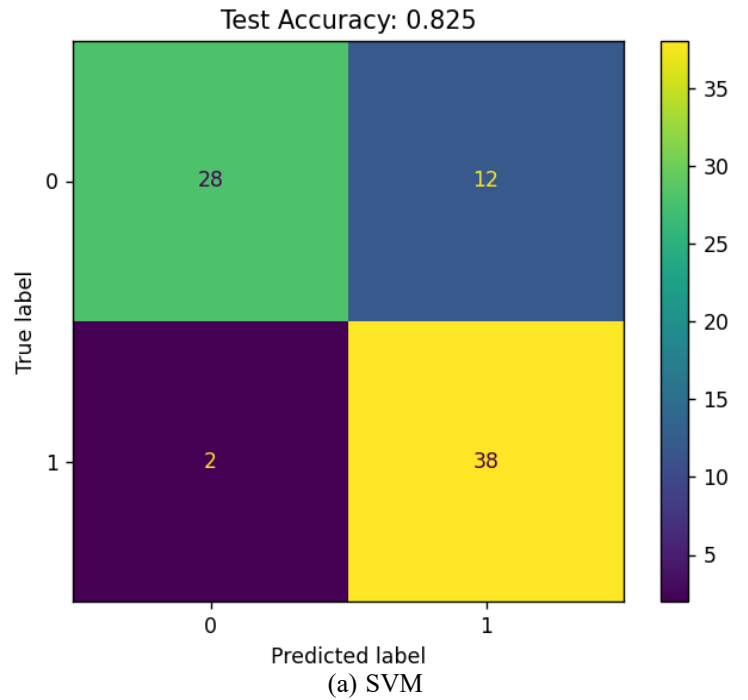


Fig. 11. Confusion matrices of optimized machine learning models.

The comprehensive hyper parameter tuning conducted on the SVM, LightGBM, and XGBoost models reveals several key insights. Firstly, fine-tuning hyper parameters significantly enhances the models' performance in classifying sound signals from the Motor Machine Sounds dataset, not only refining the models to achieve higher testing accuracy but also mitigating the risks of over fitting or under fitting observed in default configurations. The iterative nature of model refinement through hyper parameter tuning reinforces its indispensable role in achieving robust and reliable performance in real applications.

### 4.3 Machine learning model selection

This section evaluates the performance of three machine learning models - SVM, LightGBM, and XGBoost - for sound-based fault detection in machinery. Each model underwent meticulous refinement, including feature optimization and hyperparameter tuning, to maximize accuracy in classifying sound signals from the dataset. Table 8 presents a comparative analysis of these models after optimization.

Table 8. Accuracy comparison between ML models

Models	Training accuracy (%)	Testing accuracy (%)
SVM	83.12	82.50
LightGBM	90.00	81.25
XGBoost	88.75	76.25

Based on the evaluation, SVM emerges as the optimal choice for fault detection using the Motor Machine Sounds dataset. SVM demonstrates superior performance with a testing accuracy of 82.50%, closely aligned with its training accuracy of 83.12%. This minimal gap indicates excellent generalization capability, crucial for real-world applications. SVM's ability to handle high-dimensional data and maximize the margin between classes contributes to its effectiveness in this context. While LightGBM shows higher training accuracy (90.00%), its testing accuracy (81.25%) suggests a slightly higher risk of overfitting compared to SVM. XGBoost, despite its robustness, exhibits a wider gap between training (88.75%) and testing (76.25%) accuracies, indicating potential overfitting issues.

By selecting SVM, the author ensures high accuracy in classifying 'OK' and 'NG' conditions while minimizing over fitting risks. This choice lays a solid foundation for immediate application and future scalability in industrial contexts, where reliability and accuracy are paramount for effective fault detection in machinery.

### 4.4 Real application

In this section, the author assesses the performance of the optimized SVM model in real application scenarios by simulating new data classification. The testing dataset, representing 10% of the total, is divided into two scenarios: Scenario 1 and Scenario 2, as shown in Table 9. Scenario 1 has a distribution of 75% 'OK' and 25% 'NG', and Scenario 2 has the reverse distribution of 75% 'NG' and 25% 'OK'. The remaining 90% serves as the training dataset.

Table 9. Accuracy report of the scenario

Scenarios	Testing dataset distribution (%)		Accuracy (%)	
	NG	OK	Training	Testing
1	25	75	86.58	80.00
2	75	25	84.21	70

In Scenario 1, the confusion matrix in Fig. 12 shows that out of 10 instances labeled 'NG', 6 were correctly classified, and 4 were misclassified as 'OK'. Of the 30 instances labeled 'OK', 26 were correctly classified, and 4 were misclassified as 'NG'. This scenario demonstrates the model's ability to handle a majority of 'OK' labels accurately, but with some difficulty in correctly identifying 'NG' instances. The overall testing accuracy for this scenario is 80.00%, as shown in Table 9.

In Scenario 2, the confusion matrix in Fig. 13 indicates that out of 30 instances labeled 'NG', 19 were correctly classified, and 11 were misclassified as 'OK'. For 10 instances labeled 'OK', 9 were correctly classified, and 1 was misclassified as 'NG'. This scenario highlights the model's proficiency in identifying 'NG' labels, but with

an increased rate of misclassifying 'OK' labels. The overall testing accuracy for this scenario is 70.00%.

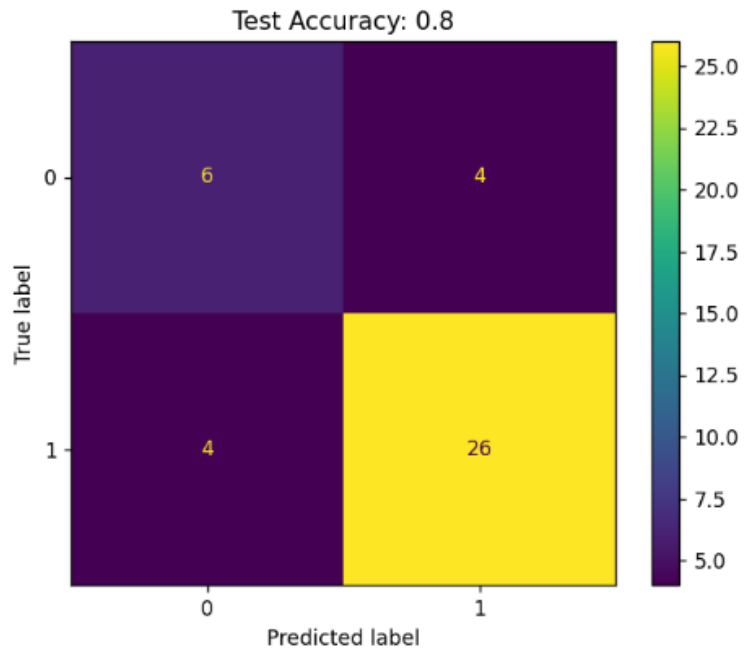


Fig. 12. Confusion matrix–scenario 1.

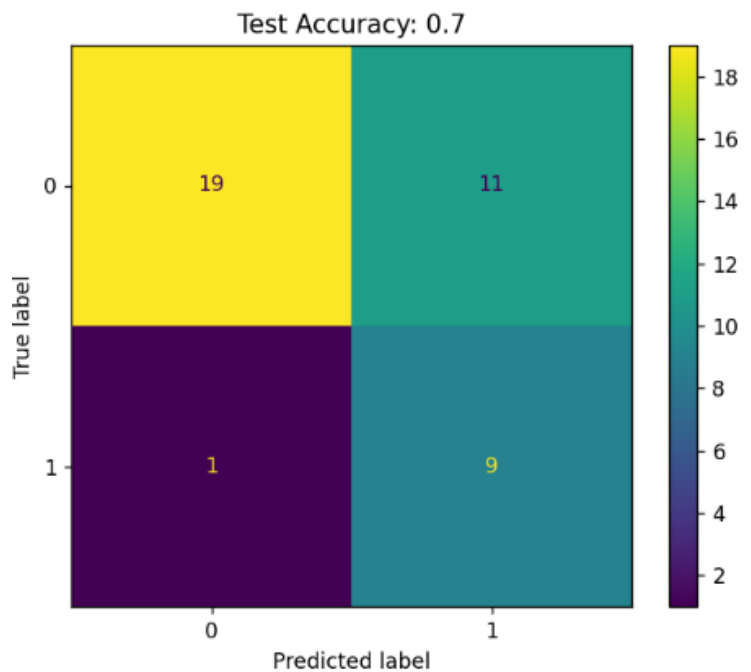


Fig. 13. Confusion matrix–scenario 2.

The drop in accuracy, particularly in Scenario 2, highlights the potential risk of over fitting, where the model struggles to maintain performance when encountering a higher proportion of 'NG' conditions. This disparity underscores the importance of balanced training data to ensure robust fault detection capabilities across various scenarios.

Imbalance data can be mitigated using several methods. The common methods are random oversampling, random under sampling of the imbalance data, or Synthetic Minority Oversampling Technique (SMOTE) [42], [43], [44]. These techniques adjust the distribution of the dataset to achieve a more balanced representation of each class. In addition, certain algorithms were designed to handle imbalances on the dataset [45]. Incorporating these techniques can lead to more robust and accurate models.

Overall, the real application simulation highlights the importance of continual model evaluation and adjustment, as initial high performance in a controlled setting does not always translate to real effectiveness. The advantages of the optimized SVM model, such as the careful MFCC feature optimization and hyper parameter tuning, have improved the model's reliability compared to its initial state.

However, further refinement and integration of additional data or techniques may be necessary to enhance the model's robustness and accuracy in practical applications, ensuring the system's reliability and effectiveness in sound-based fault detection.

## 5 Conclusions and future works

This study developed a machine fault diagnosis system using sound analysis and machine learning for damage and failure detection in industrial machinery. Acoustic features were extracted using MFCCs, with optimal performance achieved using 11 coefficients and SVM hyper parameters:  $C = 10$ ,  $\gamma = 0.1$ , and  $\text{kernel} = \text{'rbf'}$ . Among the classifiers tested (SVM, LightGBM, and XGBoost), SVM outperformed the others, achieving a training accuracy of 83.12% and testing accuracy of 82.50%. These results, along with a minimal accuracy gap, indicate strong model generalization and minimal risk of over fitting. The system effectively distinguished subtle anomalies in machine sounds, supporting its suitability for real-time, industrial applications. Furthermore, the model was validated through scenario-based simulations, demonstrating its potential for deployment in proactive maintenance strategies. Future work should address data imbalance using techniques such as random oversampling, under sampling, or SMOTE in the training and testing phases.

## Acknowledgements

This study was supported by the Faculty of Engineering and Technology (FET), the Center for Research and Community Service (CRCS), at Sampoerna University, Indonesia, and PT. Denso Manufacturing Indonesia.

## References

- [1] S. Zulaikah, F. Triawan, B. A. Budiman, and Y. Romadhon, "DEVELOPMENT OF MESSAGE CHAIR MADE OF CARDBOARD MATERIALS: DESIGN, FABRICATION, AND STRENGTH EVALUATION," *Jurnal Rekayasa Mesin*, vol. 15, no. 2, pp. 901–910, Aug. 2024, doi: 10.21776/JRM.V15I2.1593.
- [2] V. Atluri, A. Baig, and S. Rao, "Why industrials should pursue a tech-enabled transformation now," 2019.
- [3] S. Chachada and C. C. J. Kuo, "Environmental sound recognition: A survey," in *2013 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference, APSIPA 2013*, 2013. doi: 10.1109/APSIPA.2013.6694338.
- [4] W. Mu, B. Yin, X. Huang, J. Xu, and Z. Du, "Environmental sound classification using temporal-frequency attention based convolutional neural network," *Sci Rep*, vol. 11, no. 1, Dec. 2021, doi: 10.1038/s41598-021-01045-4.
- [5] J. Salamon and J. P. Bello, "Deep Convolutional Neural Networks and Data Augmentation for Environmental Sound Classification," *IEEE Signal Process Lett*, Aug. 2016, doi: 10.1109/LSP.2017.2657381.
- [6] E. C. Nunes, "Anomalous Sound Detection with Machine Learning: A Systematic Review," *ArXiv*, Feb. 2021, [Online]. Available: <http://arxiv.org/abs/2102.07820>
- [7] M. R. Ahmed, T. I. Robin, and A. A. Shafin, "Automatic Environmental Sound Recognition (AESR) using convolutional neural network," *International Journal of Modern Education and Computer Science*, vol. 12, no. 5, pp. 41–54, 2020, doi: 10.5815/ijmecs.2020.05.04.
- [8] A. Hamza *et al.*, "Deepfake Audio Detection via MFCC features using Machine Learning," *IEEE Access*, 2022, doi: 10.1109/ACCESS.2022.3231480.
- [9] R. K. K. V and Dr. S. S. Patil, "Machine Fault Detection with Sound Patterns using Deep Learning," *International Journal of Creative Research Thoughts (IJCRT)*, vol. 10, pp. 2320–2882, 2022, [Online]. Available: [www.ijcrt.org](http://www.ijcrt.org)
- [10] M. Massoudi, S. Verma, and R. Jain, "Urban Sound Classification using CNN," in *Proceedings of the 6th International Conference on Inventive Computation Technologies, ICICT 2021*, Institute of Electrical and Electronics Engineers Inc., Jan. 2021, pp. 583–589. doi: 10.1109/ICICT50816.2021.9358621.
- [11] Z. Xue, C. Xu, and D. Wen, "Structural Damage Detection Based on One-Dimensional Convolutional Neural Network," *Applied Sciences (Switzerland)*, vol. 13, no. 1, Jan. 2023, doi: 10.3390/app13010140.
- [12] K. Zhong, S. Teng, G. Liu, G. Chen, and F. Cui, "Structural damage features extracted by convolutional neural networks from mode shapes," *Applied Sciences (Switzerland)*, vol. 10, no. 12, Jun. 2020, doi: 10.3390/app10124247.
- [13] Md. A. Hossain, S. Memon, and M. A. Gregory, "A Novel Approach for MFCC Feature Extraction," *2010 4th International Conference on Signal Processing and Communication Systems*, pp. 1–5, 2010.
- [14] B. A. Budiman, H. Budijanto, F. Adziman, F. Triawan, R. Wirawan, and I. P. Nurprasetyo, "On predicting crack length and orientation in twill-woven CFRP based on limited data availability using a physics-based, high fidelity machine learning approach," *Composites Part C: Open Access*, vol. 11, p. 100371, Jul. 2023, doi: 10.1016/J.JCOMC.2023.100371.
- [15] M. Guo, Y. Guo, Y. Peng, W. Zhang, and Q. Ling, "Fault diagnosis of bolt loosening based on LightGBM recognition of sound signal features," *IEEE Sens J*, Oct. 2023, doi: 10.1109/JSEN.2023.3303223.
- [16] A. Abid, M. T. Khan, and J. Iqbal, "A review on fault detection and diagnosis techniques: basics and beyond," *Artif Intell Rev*, vol. 54, no. 5, pp. 3639–3664, Jun. 2021, doi: 10.1007/s10462-020-09934-2.
- [17] V. Kramar and V. Alchakov, "Time-Series Forecasting of Seasonal Data Using Machine Learning Methods," *Algorithms*, vol. 16, no. 5, May 2023, doi: 10.3390/a16050248.
- [18] A. K. Kemalkar and V. K. Bairagi, "Engine Fault Diagnosis Using Sound Analysis," *2016 International Conference on Automatic Control and Dynamic Optimization Techniques (ICACDOT)*, pp. 943–946, 2016.
- [19] H. Wan, X. Gu, S. Yang, and Y. Fu, "A Sound and Vibration Fusion Method for Fault Diagnosis of Rolling Bearings under Speed-Varying Conditions," *Sensors*, vol. 23, no. 6, Mar. 2023, doi: 10.3390/s23063130.
- [20] Y. Cao, Y. Sun, G. Xie, and P. Li, "A Sound-Based Fault Diagnosis Method for Railway Point Machines Based on Two-Stage Feature Selection Strategy and Ensemble Classifier," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 8, pp. 12074–12083, Aug. 2022, doi: 10.1109/TITS.2021.3109632.
- [21] F. Rustam *et al.*, "Railway Track Fault Detection Using Selective MFCC Features from Acoustic Data," *Sensors (Basel)*, vol. 23, no. 16, Aug. 2023, doi: 10.3390/s23167018.
- [22] F. Triawan *et al.*, "A quad-cliff mechanism for eco-printing by pounding technique: design, manufacturing, and testing," *Jurnal Polimesin*, vol. 22, no. 5, pp. 532–537, Oct. 2024, doi: 10.30811/JPL.V22I5.5738.
- [23] M. Fernandes, J. M. Corchado, and G. Marreiros, "Machine learning techniques applied to mechanical fault diagnosis and fault prognosis in the context of real industrial manufacturing use-cases: a systematic literature review," *Applied Intelligence*, vol. 52, no. 12, pp. 14246–14280, Sep. 2022, doi: 10.1007/s10489-022-03344-3.
- [24] P. Kumar and A. S. Hati, "Review on Machine Learning Algorithm Based Fault Detection in Induction Motors," *Archives of Computational Methods in Engineering*, vol. 28, no. 3, pp. 1929–1940, May 2021, doi: 10.1007/s11831-020-09446-w.
- [25] M. Xu, L.-Y. Duan, J. Cai, L.-T. Chia, C. Xu, and Q. Tian, "HMM-Based Audio Keyword Generation."
- [26] Z. K. Abdul and A. K. Al-Talabani, "Mel Frequency Cepstral Coefficient and its Applications: A Review," 2022, *Institute of*

- Electrical and Electronics Engineers Inc.* doi: 10.1109/ACCESS.2022.3223444.
- [27] ETSI, "Speech Processing, Transmission and Quality Aspects (STQ); Distributed speech recognition; Front-end feature extraction algorithm; Compression algorithms," Sophia Antipolis Cedex, 2003.
- [28] L. Muda, M. Begam, and I. Elamvazuthi, "Voice Recognition Algorithms using Mel Frequency Cepstral Coefficient (MFCC) and Dynamic Time Warping (DTW) Techniques," 2010.
- [29] G. Choudakkanavar, J. A. Mangai, and M. Bansal, "MFCC based ensemble learning method for multiple fault diagnosis of roller bearing," *International Journal of Information Technology (Singapore)*, vol. 14, no. 5, pp. 2741–2751, Aug. 2022, doi: 10.1007/s41870-022-00932-x.
- [30] S. Raschka, J. Patterson, and C. Nolet, "Machine learning in python: Main developments and technology trends in data science, machine learning, and artificial intelligence," Apr. 01, 2020, *MDPI AG*. doi: 10.3390/info11040193.
- [31] B. E. Boser, I. M. Guyon, and V. N. Vapnik, "Training algorithm for optimal margin classifiers," in *Proceedings of the Fifth Annual ACM Workshop on Computational Learning Theory*, Publ by ACM, 1992, pp. 144–152. doi: 10.1145/130385.130401.
- [32] C. Cortes, V. Vapnik, and L. Saitta, "Support-Vector Networks Editor," Kluwer Academic Publishers, 1995.
- [33] G. Ke *et al.*, "LightGBM: A Highly Efficient Gradient Boosting Decision Tree," *Advances in Neural Information Processing Systems 30 (NIP 2017)*, 2017, [Online]. Available: <https://github.com/Microsoft/LightGBM>.
- [34] M. Roberto Machado, S. Karray, and I. T. de Sousa, "LightGBM: an Effective Decision Tree Gradient Boosting Method to Predict Customer Loyalty in the Finance Industry," *The 14th International Conference on Computer Science & Education (ICCSE 2019)*, 2019, Accessed: Nov. 14, 2023. [Online]. Available: <https://ieeexplore.ieee.org/document/8845529>
- [35] M. J. Sai, P. Chettri, R. Panigrahi, A. Garg, A. K. Bhoi, and P. Barsocchi, "An Ensemble of Light Gradient Boosting Machine and Adaptive Boosting for Prediction of Type-2 Diabetes," *International Journal of Computational Intelligence Systems*, vol. 16, no. 1, Dec. 2023, doi: 10.1007/s44196-023-00184-y.
- [36] T. Chen and C. Guestrin, "XGBoost: A Scalable Tree Boosting System," Mar. 2016, doi: 10.1145/2939672.2939785.
- [37] C. Bentéjac, A. Csörgő, and G. Martínez-Muñoz, "A Comparative Analysis of XGBoost," Nov. 2019, doi: 10.1007/s10462-020-09896-5.
- [38] X. Chen, S. Wang, B. Qiao, and Q. Chen, "Basic research on machinery fault diagnostics: Past, present, and future trends," Jun. 01, 2018, *Higher Education Press*. doi: 10.1007/s11465-018-0472-3.
- [39] W. Mustikarini, R. Hidayat, and A. Bejo, "Real-Time Indonesian Language Speech Recognition with MFCC Algorithms and Python-Based SVM," 2019.
- [40] R. Guido, M. C. Groccia, and D. Conforti, "A hyper-parameter tuning approach for cost-sensitive support vector machine classifiers," *Soft comput*, vol. 27, no. 18, pp. 12863–12881, Sep. 2023, doi: 10.1007/s00500-022-06768-8.
- [41] S. Li, N. Jin, A. Dogani, Y. Yang, M. Zhang, and X. Gu, "Enhancing LightGBM for Industrial Fault Warning: An Innovative Hybrid Algorithm," *Processes*, vol. 12, no. 1, Jan. 2024, doi: 10.3390/pr12010221.
- [42] B. W. Yap, K. A. Rani, H. A. Abd Rahman, S. Fong, Z. Khairudin, and N. N. Abdullah, "An Application of Oversampling, Undersampling, Bagging and Boosting in Handling Imbalanced Datasets," *Lecture Notes in Electrical Engineering*, vol. 285 LNEE, pp. 13–22, 2014, doi: 10.1007/978-981-4585-18-7\_2.
- [43] T. Wongvorachan, S. He, and O. Bulut, "A Comparison of Undersampling, Oversampling, and SMOTE Methods for Dealing with Imbalanced Classification in Educational Data Mining," *Information 2023, Vol. 14, Page 54*, vol. 14, no. 1, p. 54, Jan. 2023, doi: 10.3390/INFO14010054.
- [44] L. Zhang *et al.*, "Classification of Imbalanced Data: Review of Methods and Applications," *IOP Conf Ser Mater Sci Eng*, vol. 1099, no. 1, p. 012077, Mar. 2021, doi: 10.1088/1757-899X/1099/1/012077.
- [45] H. Hassan, N. B. Ahmad, and S. Anuar, "Improved students' performance prediction for multi-class imbalanced problems using hybrid and ensemble approach in educational data mining," *J Phys Conf Ser*, vol. 1529, no. 5, p. 052041, May 2020, doi: 10.1088/1742-6596/1529/5/052041.