Intelligent Fault Diagnosis in Industrial Machinery: Leveraging AI with LSTM Autoencoder for Enhanced Fault Detection

[1]Rupa Devi B, [2]Suseela G, [3]Ranjith Kumar Painam, [4]Thammisetty Swetha, [5]Suryanarayana G and [6]Reddy Madhavi K
[1]Department of Computer Science and Engineering, Annamacharya Institute of Technology and Sciences, Tirupati, Andhra Pradesh, India.
[2]Department of Networking and Communications, College of Engineering and Technology, School of Computing, SRM Institute of Science and Technology, Kattankulathur, Chengalpattu, Tamil Nadu, India.
[3]Department of Electronics and Communication Engineering, Kallam Haranadhareddy Institute of Technology (Autonomous), Guntur, Andhra Pradesh, India.
[4]Department of Computer Science and Engineering - Data Science, Madanapalle Institute of Science & Technology, Madanapalle, Andhra Pradesh, India.
[5]Department of Information Technology, Vardhaman College of Engineering, Shamshabad, Hyderabad, Telangana, India.
[6]Department of AI & ML, School of Computing, Mohan Babu University, Tirupati, Andhra Pradesh, India.
[1]rupadevi.aitt@annamacharyagroup.org, [2]suseelag@srmist.edu.in, [3]ranjithkumar.painam@gmail.com, [4]swethathammisetty7@gmail.com, [5]surya.aits@gmail.com, [6]kreddymadhavi@gmail.com

Correspondence should be addressed to Ranjith Kumar Painam: ranjithkumar.painam@gmail.com

Article Info
Journal of Machine and Computing (http://anapub.co.ke/journals/jmc/jmc.html)
Doi : https://doi.org/10.53759/7669/jmc202404086
Received 20 January 2024; Revised from 10 April 2024; Accepted 20 July 2024.
Available online 05 October 2024.

Abstract – Machinery Fault Detection (MFD) is an important process in contemporary industrial systems, where it predicts possible physical failures before they lead to a serious problem. This uses multiple technologies to monitor machine statuses (algorithms, data gathering systems and sensors) Using a servo-motor driven actuator for deployment, the Locking Mechanism is pre-assembled into an OEM ATE and will enable predictive failure mode identification (via monitoring and warnings of operational parameters i.e., vibration, temperature or auditory signals in-built to MFD systems) leading to Prophylactic maintenance before critical bottlenecks can occur. The dataset we used in our study was collected from Kaggle and it is called the SpectraQuest Machinery Fault Simulator (MFS) Alignment-Balance-Vibration (ABVT). We used LSTM Autoencoder, KNN, SVM and DNN to analyzed the data. Our LSTM Autoencoder model was very accurate and achieved a precision, recall, accuracy and F-score of 99%. We worked on very large scale datasets. It will help the system detect faults and predict their evolution over time, so you save maintenance costs and increase production in your factory. More research on the practical efficiency of these models in real-time across different industrial settings can create a path towards improved and scalable MFD solutions.

Keywords – Machinery Fault Detection (MFD), Industrial Maintenance, Machine Learning in Fault Detection, Fault Detection, Deep Learning, LSTM Autoencoder.

I. INTRODUCTION

Machinery fault detection (MFD) is very important when it comes to guaranteeing the performance, cost effectiveness and operational life of most modern industrial equipment [1]. Electricity is used in many places from industry to commerce and at home; most of that require electrical machinery. However, these machines work under very harsh conditions-high temperatures hard work fluctuations unstable voltages high humidity all of them increase malfunction -which may cause failure [2]. Studies show that 30–40% of faults in rotating electrical machines are due to bearing failures; and poor lubrication accounts for about80%, being the most common root cause. In today's automated manufacturing environments, robotic systems and multi-axis machines are essential for improving efficiency in welding, packaging, material transport, and assembly [3]. Yet, mechanical faults become more common as industries push machinery to work under extreme conditions, such as heavy loads and high humidity. These failures can disrupt operations, result in expensive repairs, reduce output, and even create safety hazards. Therefore, preventive maintenance and early defect identification are crucial for reducing downtime and preserving efficient operations. As industrial technology continues to develop rapidly, the reliability of machinery systems and components has become increasingly significant, fostering the growing integration of information technology with industrial operations. Highly reliable systems are essential to operational safety in the aircraft sector. Due to its prolonged exposure to high loads, harsh temperatures, and unfavorable conditions, spinning machinery—such as that found in space shuttles and aviation engines—is particularly prone to breakdowns. Early problem identification is crucial to preventing mechanical transmission system failures because of the possible consequences of any fault in these systems. Therefore, industries prioritize sophisticated fault diagnosis systems for rotating machinery to lower the likelihood of minor issues becoming significant breakdowns [4]. Scholars have concentrated on the revolving machinery that is embodied in aviation gear. Early fault diagnosis and detection technologies are essential for preventing transmission system failures in mechanical engineering because they may predict the pattern of fault development. Thus, every industry places a high value on the intelligent fault diagnosis system on rotating machinery in order to prevent further serious accidents brought on by little flaws [5].

The significance of MFD has been widely recognized by both academic and industrial sectors, leading to the development of progressive diagnostic methods for practical use. Traditional maintenance techniques, such as preventive or corrective maintenance, often require identifying faults early, resulting in inefficiencies and unexpected breakdowns. Recent advancements in sensor technology, data collection systems, and Artificial Intelligence (AI) have revolutionized fault detection. These developments have transformed maintenance from reactive to predictive. Modern systems can frequently monitor indicators like vibrations, temperature, and acoustic emissions in real time to identify irregularities and anticipate potential equipment malfunctions. The field of MFD is increasing, leading to quick growth in research and discoveries [6]. However, existing studies of Machine Learning (ML) and Deep Learning (DL) applications in MFD often need to be more cohesive, leading to critical gaps in the field's comprehensive understanding. Consequently, of this evolution, a wave of new research has appeared to improve the interpretation and application of these techniques. These developments overcome existing gaps and strengthen the robustness of fault diagnosis capabilities by refining the encoding, processing, and prediction of fault information inside ML/DL frameworks.

The scarcity of research concentrating on individual MFD underscores the novelty and significance of our work. While much of the existing literature addresses broader or generalized fault detection across different machinery types, our study explicitly targets individual machinery components employing an advanced LSTM Autoencoder model. The significance of our research lies in its advancement of MFD using sophisticated ML techniques by utilizing a Kaggle and our study provides necessary insights into the effectiveness of different algorithms in diagnosing and predicting machinery faults. We will assess the effectiveness of K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Deep Neural Networks (DNN), and LSTM Autoencoder for fault detection and prediction. Furthermore, we provide practical recommendations for improving maintenance schedules, cutting operational expenses, and improving equipment dependability and safety. This focused approach allows for more accurate and effective fault detection in specific machinery contexts, addressing current research gaps and presenting valuable insights for targeted fault detection strategies.

This paper is divided into five sections. Section II will examine the current techniques for detecting machinery faults. Next, Section III will outline our proposed approach and briefly cover the setup for our model's implementation. Section IV will summarize the results of the experiments using the models we have offered. Lastly, we will present our research findings.

II. LITERATURE REVIEW

The conventional fault detection (FD) methods used in industrial machinery are thoroughly reviewed in the following section, emphasizing the benefits and drawbacks of each methodology. For many years, conventional methods have been the cornerstone of fault detection in industrial machines and robots. The term "fault detection" describes analyzing and classifying defects in real-time diagnostics that happen after the fact. Any unacceptably significant deviation in the system that affects sensors, actuators, and other components is a problem. The two main types of FD methods used in machinery or processes are model-based approaches and data-driven methods.

Several existing studies already cover ML and DL-based approaches to MFD, each concentrating on distinct components or equipment such as bearings, gears, pumps, or motors. Zhang et al. [7], Neupane et al. [8], Hoang et al. [9], and Mushtaq et al. [10] provided summaries of ML and DL methods for various machinery components. However, Srilatha et al. [11] and Tang et al. [12] have examined current techniques for DL-based MFD and emphasized the significance of data processing in condition monitoring. Similarly, Zhang et al. [13] looked at methods for dealing with unbalanced and short datasets and offered fixes, including feature learning, data enrichment, and classifier design. Zheng et al. [14] and Li et al. [15] investigated MFD domain adaptation and transfer learning. Additionally, Autoencoder (AE)--based techniques were examined by Yang et al. [16] and Qian et al. [17], who highlighted the advantages and disadvantages of AE in industrial applications. Zhu et al. [18] summarized the use of Recurrent Neural Networks (RNNs) in mechanical fault diagnostics. Murugan et al. [12] and Tang et al. [19] offered more information on the basic principles of CNNs for rotating equipment failure detection.

A thorough analysis of several fault detection and isolation techniques was provided by Kim et al. [20], with an emphasis on essential elements, including affordability, adaptability, durability, precision, comprehensibility, and simplicity of use. Because of their ease of use, threshold- and rule-based systems are frequently acknowledged for being highly economical. However, model-based FDI and acoustic emission (AE) monitoring may be more costly because of the difficulty of creating the model and the need for specific sensors. Regarding adaptability, rule-based systems and signal-processing approaches are better suited to various machines and failure circumstances.

Mechanical systems are intricate, but the identification of faults can be significantly improved through methods such as data augmentation, processing data at multiple scales, and achieving fault signals from various sensors. A specific multi-scale CNN was presented by An et al. [21] as a means of diagnosing rolling bearing problems under challenging circumstances such as high noise levels, fluctuating loads, and intricate workplace environments. This method successfully raises network resilience and diagnostic accuracy. Likewise, Li et al. [22] overcame the difficulties in extracting and identifying failure features for electromechanical equipment by developing a fault diagnosis technique based on a multi-scale, one-dimensional deep convolutional network. Their approach demonstrated remarkable robustness, high rates of defect identification, and great diagnostic accuracy. Furthermore, Hasan et al. [23] presented a method for identifying bearing problems that combine data from several sensors to overcome the difficulty of identifying internal and exterior multi-excitation problems in aero-engine bearings. This methodology demonstrated a higher capacity to recognize problem types properly, increasing fault diagnostic accuracy by 36.92% and 18.9%, respectively, compared to established methods like SVM and ANN.

Despite their shortcomings, conventional techniques are vital for identifying faults with robotics and industrial machinery. They offer an efficient, low-cost method of monitoring crucial system parameters and locating faults. Nonetheless, as AI and ML advances allow for greater accuracy and early mistake detection, they are becoming increasingly more beneficial for foreign direct investment.

III. METHODOLOGY

Data Preprocessing

There were several important steps executed during the data preprocessing phase of our study, supported with mathematical modeling to shape up dataset in an appropriate manner for proper machine learning exercise. We started with down sampling, which meant averaging data points into fixed partitions to decrease the size of a set but preserving vital patterns. To do this we simply added the data in each interval and divided by the size of the interval while keeping our key properties. Next, we used MinMax scaling to normalize the downsampled data and map features to a range of [-2,2] using the transformation $f_i^1 = \frac{fi\ min\ (f)}{(f)\ -(f)} * 4$. In this way, all features played an equal part in learning the model. We reshaped the data so that it fits to input required by lstm autoencoder which means we changed our feature matrix X from a shape of (m, n) (m,n)–>to –>(m, 1, n) adding an essential time-step dimension for sequence modeling. Finally, using the train_test_split function to split this data that at a ratio of 98:2, so we could know how real model performed with unseen test dataset. The improvements in the model accuracy that we have gained is by following these preprocessing techniques are so important and crucial as it helps to correctly identify which of machine has a problem. The shape of the Matrix was (12,005,000,3), since our training data had 12 samples and each sample having three features. This large data set was used to train the model, which fortify and help develop blanket support base required for ascertaining underlying problems connected with machine and its functioning. The testing data included 245,000 samples with three attributes, yielding a shape of (245,000,3). This smaller section was set aside for testing the model's performance, allowing the accuracy and robustness of the model to be investigated using previously unreported data.

Autoencoder

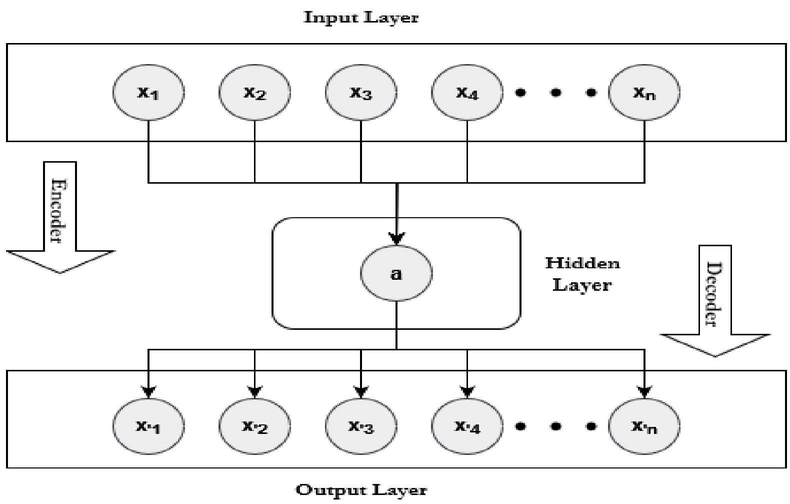

Fig 1. A Single Autoencoder Model Architecture.

An ANN called an autoencoder creates output vectors that are comparable to the inputs by using backpropagation. Initially the technique uses a lower-dimensional space to compress the input data, and then it employs this representation to recover the original data. The technique utilizes many layers and a non-linear activation function to detect nonlinear correlations within the data.

Fig 1 depicts an illustration of the architecture of the autoencoder model. Since the Autoencoder requires not a separate label value to be trained, it is categorized as an unsupervised learning method. The encoder and decoder sections comprise the two stages of the autoencoder. Equation (1) explains the encoder phase's goal of minimizing the dimensions of input data X.

$$A = \sigma (BX + c) \quad (1)$$

A represents the latent dimension, σ stands for the activation functions, B for weight, and x for the bias vector. Equation (2) is used to train the decoding phase to provide output data that is exactly the same as the original space but has different activation functions (σ), weight B, and bias x.

$$X' = \sigma' (B'A + c') \quad (2)$$

The goal of the autoencoder is to reduce the amount of reconstruction error between the original vectors and the output. Either the sum of squared errors (SSE) or a cross-entropy function can be used to compute the reconstruction error. In this study, we computed the reconstruction error utilizing equation (3) through SSE.

$$SSE = \sum_{i=1}^{n} (X'_i - X_i)^2 \quad (3)$$

The decoder generates the starting data based on the encoder output. The core layer at the centre of the autoencoder structure reduces the dimensionality of input data and generates a compressed feature vector. The coding layer can be classified or integrated with other stacked autoencoders.

LSTM

The traditional RNN architecture struggles to handle recurring data. Conventional RNNs struggle to learn long-term dependencies. Training is challenging due to the exponential degradation of outputs caused by information being fed back into the hidden layer at every time step. The "exploding gradient problem" or the "vanishing gradient problem" is what this is known as. One potential solution to address these issues is to use the well-known "Long-Short-Term Memor" (LSTM) architecture. Due of its ability to manage long-term dependencies and maintain gradient information over time, the LSTM architecture is chosen over standard RNN designs. Each RNN neuron becomes a "block" when the LSTM adds numerous operations to it. A typical RNN block is shown in **Fig 2**. The basic block in **Fig 3** can be expanded in multiple ways using the LSTM. The "cell state" x(t) in R, where x denotes the state's dimensionality, is initially created. Information can be stored in this state for future use if necessary. Here, "memory" refers to the LSTM block. This is followed by sending the concatenated input x(t) and the previous output through a series of built "gates". Applying learning weights and biases Wf, Wi, Wo, and bf, bi, bo to the input z(t − 1), the sigmoid function generates an x-dimensional vector with values in the range [0, 1]. Non-linear functions, like tanh, are represented by g and h, while element-wise addition and multiplication are expressed by L and N.

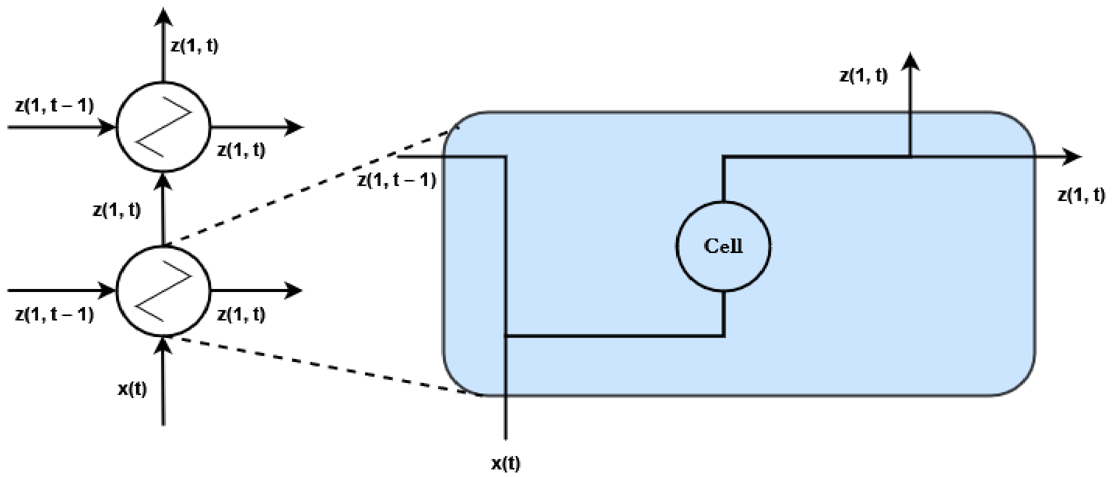

Fig 2. A Standard RNN Block.

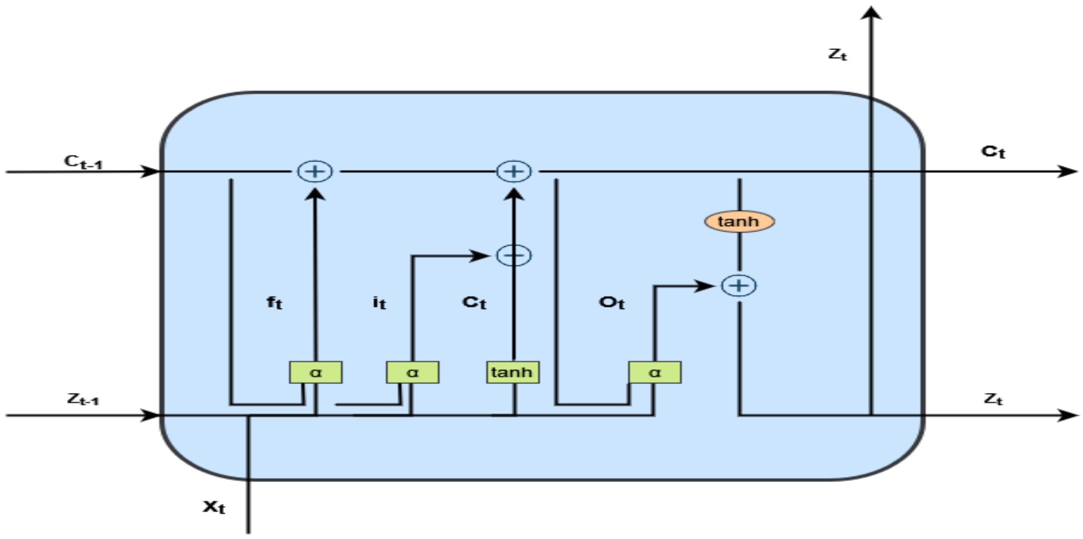

Fig 3. Graphical Representation of LSTM Block.

Proposed Model

The architecture of the LSTM-based autoencoder for machinery fault detection can be described through its encoder and decoder components alongside the associated equations. The input to the model is a time-series sequence A = $\{\alpha_{1,}\alpha_{2,}\ldots\ldots,\alpha_T\}$, where each $\alpha_t \in R^n$ represents a vector of features at timestep t, with T being the complete timesteps count and n the number of features. The encoder compresses this input sequence into a lower-dimensional latent vector z, capturing the critical temporal patterns. This is achieved through two LSTM layers. The first LSTM layer generates hidden states –

$$h_t^{(1)} = (LSTM_1\,(\alpha_{t,}h_{t-1}^{(1)}, c_{t-1}^{(1)}) \tag{4}$$

where $h_t^{(1)}$ and $c_t^{(1)}$ are the hidden and cell states at time t, respectively. The second LSTM layer refines these states to produce –

$$h_t^{(2)} = (LSTM_2\,(h_t^{(1)}, h_{t-1}^{(2)}, c_{t-1}^{(2)}) \tag{5}$$

with $h_t^{(2)}$ and $c_t^{(2)}$ representing the hidden and cell states for this layer. The final hidden state $h_T^{(2)}$ at timestep, T serves as the encoded representation z ∈ R^d, where d is the dimension of the latent space. In the decoder, this latent vector z is repeated T times to match the sequence length and is processed through two additional LSTM layers, similar to the encoder. The third LSTM layer produces hidden states –

$$h_t^{(3)} = (LSTM_3\,(Z_t, h_{t-1}^{(3)}, c_{t-1}^{(3)}) \tag{6}$$

followed by the fourth LSTM layer which refines these hidden states as,

$$h_t^{(4)} = (LSTM_4\,(h_t^3, h_{t-1}^{(4)}, c_{t-1}^{(4)}) \tag{7}$$

A dense layer then reconstructs the original sequence by mapping each $h_t^{(4)}$ to the output feature vector $\alpha_t = Dense\ h_t^4$, resulting in the reconstructed sequence A = $\{\alpha_{1,}\alpha_2,\ldots\ldots,\alpha_T\}$. The model is trained by decreasing the mean squared error (MSE) between the original and reconstructed sequences, defined as,

$$\mathrm{MSE} = \frac{1}{nT}\sum_{i=1}^{T}\sum_{j=1}^{n}(\alpha_{ij} - \alpha\hat{}_{ij})^2 \tag{8}$$

where α_{ij} and $\alpha\hat{}_{ij}$ are the original and reconstructed values for feature j at timestep i. The reconstruction error, captured by the MSE, is utilized to classify the machine's health status, with a low MSE indicating a healthy state, a medium MSE suggesting a degraded state, and a high MSE signaling a potential fault. This approach enables early detection of machinery faults, allowing for timely maintenance interventions.

IV. EXPERIMENTAL SETUP

Dataset Description

This dataset, fetched from Kaggle, comprises 1,951 multivariate time-series samples gathered employing the SpectraQuest Machinery Fault Simulator (MFS) Alignment-Balance-Vibration (ABVT). It incorporates data for six operational states: Normal function, Imbalance faults, Horizontal and vertical misalignment faults, and Inner and outer bearing faults.

Every sequence is captured over 5 seconds at a sample rate of 50 kHz.

Key Fault Conditions

- Normal Operation: 49 sequences
- Imbalance Faults: 333 sequences with weights ranging from 6 g to 35 g
- Horizontal Misalignment: 197 sequences with shifts from 0.5 mm to 2.0 mm
- Vertical Misalignment: 301 sequences with shifts from 0.51 mm to 1.90 mm
- Bearing Faults: 558 sequences for underhang bearings and 513 sequences for overhang bearings, addressing issues with rolling components, inner tracks, and exterior tracks.

The data is obtained using high-precision accelerometers, a tachometer, a microphone, and an analog acquisition module. **Table 1** and **2** illustrate vital aspects of the experimental setup and dataset composition.

Table 1. Specifications of the Experimental Bench

Specification	**Value**	**Unit**
Motor	1/4	CV DC
Frequency range	700-3600	rpm
System weight	22	kg
Axis diameter	16	mm
Axis length	520	mm
Rotor	15.24	cm
Bearings distance	390	mm
Number of balls	8	
Balls diameter	0.7145	cm
Cage diameter	2.8519	cm
FTF	0.3750	CPM/rpm
BPFO	2.9980	CPM/rpm
BPFI	5.0020	CPM/rpm
BSF	1.8710	CPM/rpm

Table 2. Summary of Each Type of Sequence

Sequence	**Measurements**
Normal	49
Horizontal misalignment	197
Vertical misalignment	301
Imbalance	333
Underhang bearing	
Cage fault	188
Outer race	184
Ball fault	186
Overhang bearing	
Cage fault	188
Outer race	188
Ball fault	137
Total	**1951**

Experimental Setup

All models were trained with Tensorflow, a Python-based deep learning toolset. Tensorflow enables the transparent execution of incredibly efficient mathematical procedures on GPUs. A computational graph defines all necessary activities for the specified calculations. Due to a lack of adequate assessment during the project, speedup may have changed. AutoEncoder and LSTM models minimize validation loss. Adam employs 128 mini-batches and a learning rate of 0.0001 for stochastic gradient descent strategy. **Table 3** displays the model specifications and experimental summary.

Table 3. Summary of the Model Experiment.

Parameter	Value
Optimizer	Adam
Loss	Mae
Activation	Relu
Regulizer	L2
Epoch	100
Batch Size	128
Trainable params	249219
Non-trainable params	0

Evaluation Metrics

Precision

Precision measures how accurate the model's positive predictions are. It is computed using:

$$\text{Precision} = \frac{TP}{TP+FP} \tag{9}$$

Recall

Recall estimates the model's capacity to properly identify all relevant examples, or, in this case, all genuine problems. It is calculated by:

$$\text{Recall} = \frac{TP}{TP+FN} \tag{10}$$

F-score

The mean of precision and recall is the F-score (also known as the F1-score), which delivers a balanced assessment that accounts for both false positives and false negatives. This is computed by:

$$\text{F-score} = 2* \ \frac{Precision*Recall}{Precision+Recall} \tag{11}$$

Accuracy

The simplest statistic is accuracy, this represents the proportion of genuine positives and true negatives to all predictions. This ratio indicates how accurate the model is overall. It is provided by:

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FNl} \tag{12}$$

Outcomes of the Models

Table 4. Training, Testing, and Validation Accuracy of Different Models

Model	Training Accuracy (%)	Testing Accuracy (%)	Validation Accuracy (%)
KNN	92	88	87
SVM	93	89	89
DNN	95	93	92
LSTM Autoencoder	99	98	99

In this study, we compared four models' training, testing, and validation accuracies for detecting machinery faults, shown in **Table 4**: KNN, SVM, DNN, and LSTM Autoencoder. The KNN model obtained 92% training accuracy, 88% testing accuracy, and 87% validation accuracy. Although KNN is straightforward, its poor performance, particularly in testing and validation, shows limited generalization possibilities for complicated defect detection applications. Training Accuracy: 93% Testing and Validation Accuracy: SVM model training performance was slightly better with a Training accuracy of 93%, testing, validation both were at an accuracy rate of 89%. The similarity in the proportions reflects a balance between model complexity and generalization, which can be interpreted as an SVM being a reliable replacement for defect identification. At the same time, it may miss some subtle patterns in the data. The DNN model randomly given much better score of 95% training accuracy, 93% testing and also91 % validation accuracy. This illustrates the increased power of DNN to model data with non-linear relationships fully, leading to better generalization across datasets. Nevertheless, it was the LSTM Autoencoder that performed best of all–training accuracy 99%, testing accuracy 98% and validation accuracy after just two epochs. As shown in the above performances scores, high precision of LSTM

Autoencoder at all phases indicated that it can learn and reproduce temporal patterns effectively making them suitable for time-series industrial data like machinery defect detection.

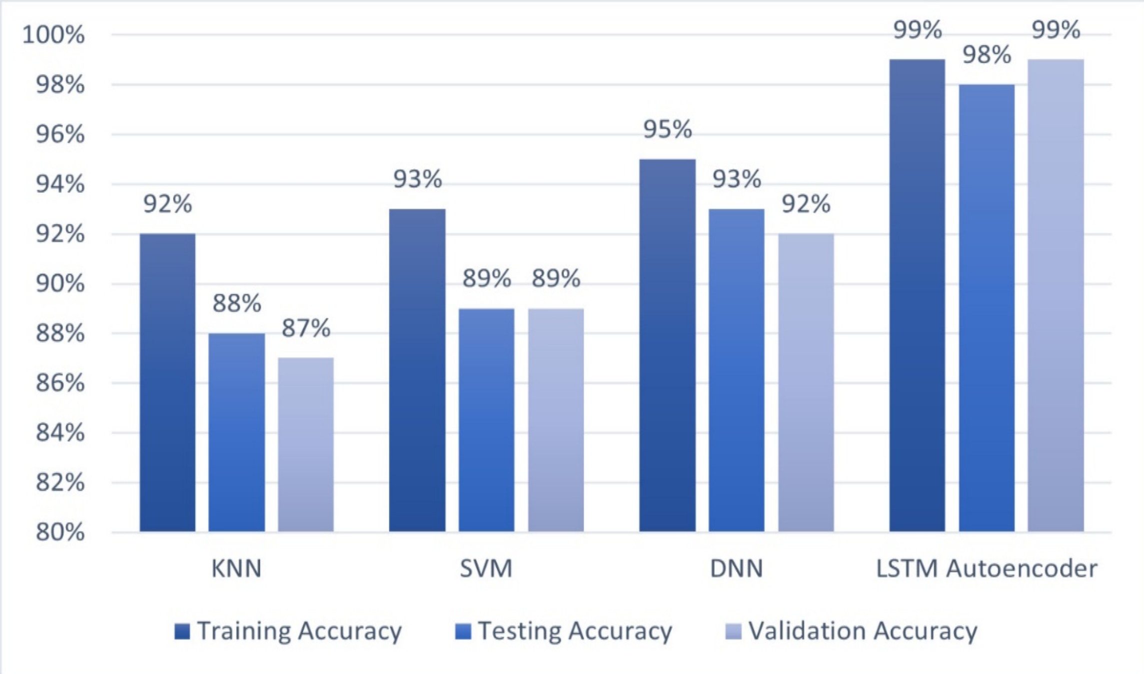

Fig 4. Training, Testing and Validation Accuracy of All Models.

Fig 4 in our study examines more comprehensively all the models explored based on training, testing and validation accuracy. This is a plot of the performance of each LSTM Autoencoder, KNN (K-nearest neighbors), SVM and DNN model in training phase, testing phase (how well they generalize to new data) and validation set(each how consistently). Indeed, the LSTM Autoencoder was extremely accurate across all three phases and generated almost perfect results since it is such an effective fault detection method. **Fig 4** graphically represents both positive and negative sides of each model; it provides a qualitative view on the models to other researchers that whether can generalize well or not, as well as whether are reliable enough on diagnosing faulty equipment.

Following the results, **Fig 5** and **6** graphically depict the training and validation curves and the associated loss metrics. This image demonstrates the model's learning dynamics across multiple epochs and the training process's efficacy and convergence patterns. The consistency between the training and validation curves, in particular for the LSTM Autoencoder, demonstrates the model's greater generalization potential and enhances its robustness in precisely recognizing machinery faults across multiple datasets.

Table 5. Precision, Recall, and F1-Score of Individual Models

Model	Precision (%)	Recall (%)	F1-Score (%)
KNN	89	87	88
SVM	90	89	89
DNN	93	92	93
LSTM Autoencoder	99	99	99

Several evaluation measures were employed to further analyze the models' performance in detecting machinery defects. A more detailed knowledge of the models' advantages and weaknesses can be obtained by comparing the MFD models based on precision, recall, and F1-score, as shown in **Table 5**. With an F1-score of 88%, recall of 87%, and accuracy of 89%, the KNN model shows that, while it can effectively recognize machine conditions, it struggles with borderline cases. The model's relatively low recall indicates that it ignores certain problematic conditions, increasing the likelihood of false negative outcomes. This means that, while KNN may be useful for easier-to-understand datasets, it may not effectively reflect the complexity involved in machinery fault identification. The SVM model performs better, achieving 90% precision and 90% recall. This improvement comes from SVM's ability to determine the optimal borders between classes, which boosts its dependability when determining a machine's healthy and malfunctioning states. Even if SVM performs better than KNN in this specific situation, the F1-score of 89% indicates that SVM is still not ideal, especially when there are noisy or overlapping classes. The DNN model gets 93% F1 score, 92% recall, and 93% precision, considerably improving performance.

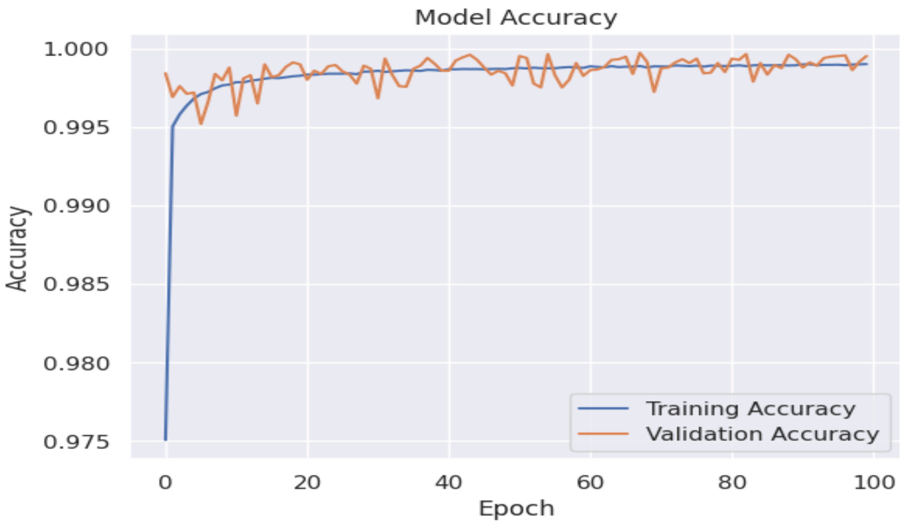

Fig 5. Training and Validation Curves Over Epochs.

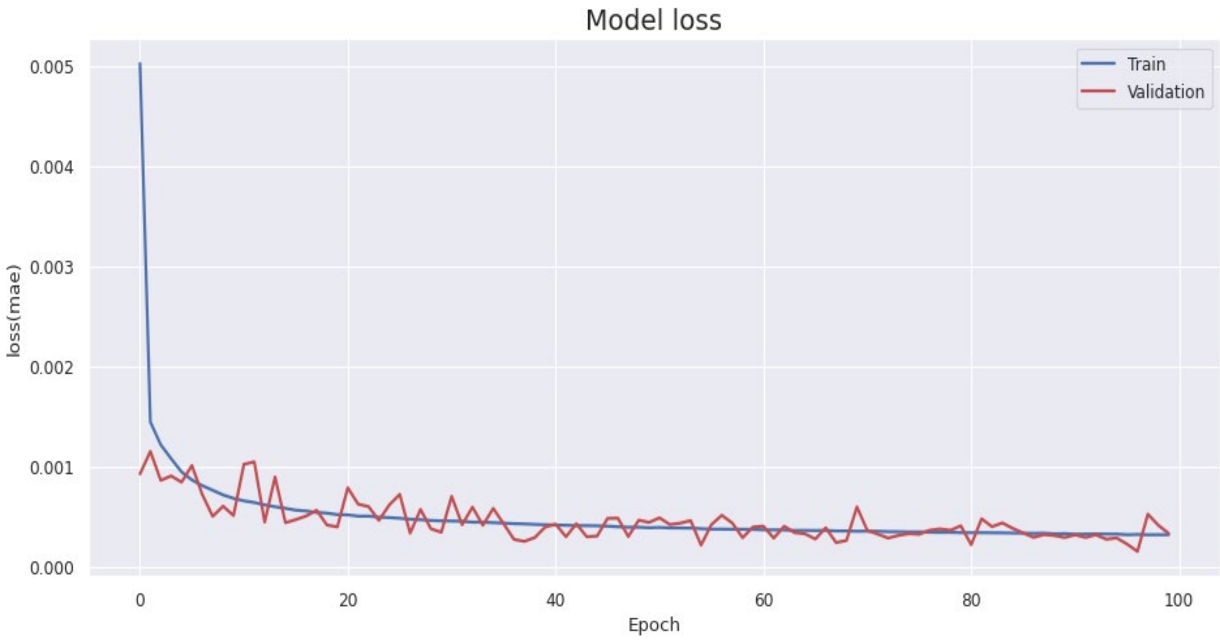

Fig 6. Training and Validation Loss Over Epochs.

This suggests that because the DNN is much better at figuring out complex, non-linear relationships from the data, it can identify errors with more accuracy. Its strong memory, which indicates that it successfully records real positives, and higher precision, which points to fewer false positives, make it a good choice for this task. Though DNN models perform better than KNN and SVM models, they might still struggle with successful generalization. This is especially true if the model is not properly calibrated or if the training data does not accurately reflect all possible fault scenarios. Compared to the other models, the LSTM Autoencoder performs far better, with F1 scores, recall, and precision all at 99%. Sequential data is very well used in LSTM development since temporal linkages are essential for detecting machinery problems. Its excellent precision shows that it can identify fault conditions with minimal false positives, and its near-perfect recall ensures that almost all defective circumstances are correctly diagnosed.

The F1-score validates the outstanding balance between the two metrics in the model. This demonstrates the accuracy and dependability of the LSTM Autoencoder across a range of operational conditions. This comprehensive analysis shows that the LSTM Autoencoder is the most effective model for identifying machine problems, particularly in situations where temporal data is critical to assessing the health of the machine. Because of its remarkable accuracy and memory, which show that it can generalize across various defect kinds and operating conditions, it is a crucial component of predictive maintenance plans. Conversely, while KNN and SVM provide a solid foundation, it is clear that they cannot handle complex data structures. Despite its strength, the DNN can still not analyze time-series data with the same accuracy and dependability as the LSTM Autoencoder.

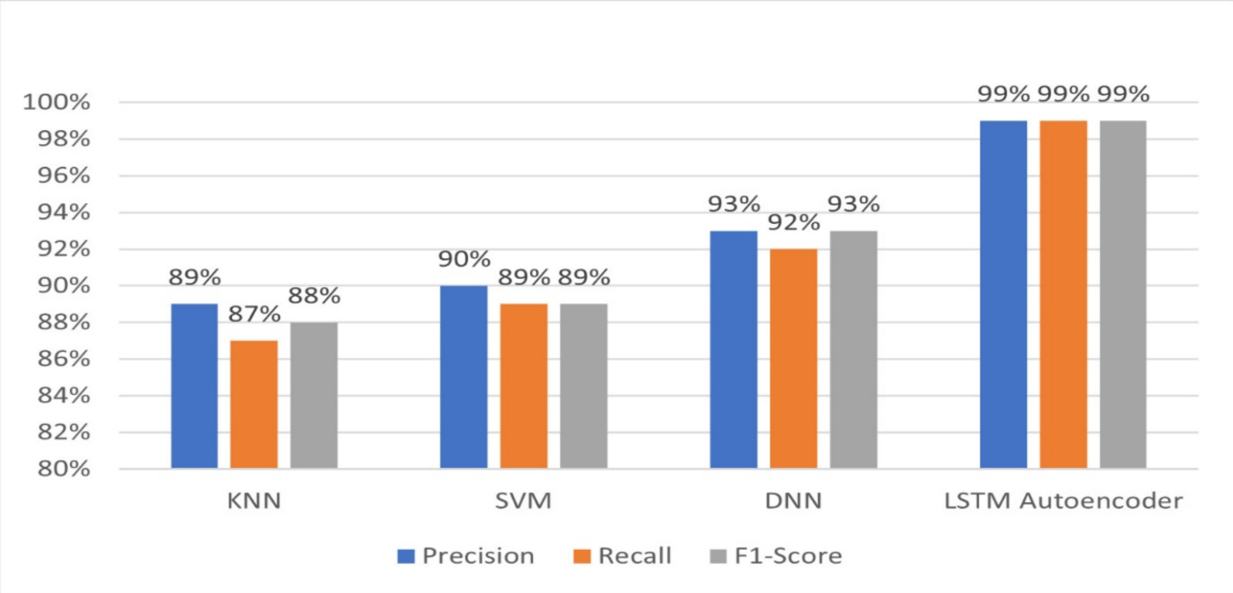

Fig 7. Precision, recall and f-score of all models

Fig 7 of our study provides a full comparison of the assessment metrics we utilized for each of the models that were evaluated: LSTM Autoencoder, KNN, SVM, and DNN. This image efficiently demonstrates how well each particular model performs in terms of remembering all relevant fault scenarios (recall), accurately identifying equipment flaws (accuracy), and striking a balance between these two attributes (F-score). The LSTM Autoencoder stands out as having a stronger ability to precisely identify flaws and decrease false positives and negatives, obtaining the highest scores in all three criteria.

Discussion

In the evolving field of MFD, acquiring high accuracy and efficiency remains a crucial challenge. Recent advancements have presented various models to improve fault detection capabilities across industrial applications. A discussion is conducted to show the effectiveness of our proposed LSTM Autoencoder model in detecting machinery faults, comparing its performance with existing approaches. **Table 6** outlines the objectives, models, and accuracy of different fault detection approaches, including our suggested LSTM Autoencoder model, showing its superior performance compared to existing methods.

Table 6. Comparative Performance of Fault Detection Models

Reference	Objective	Model	Accuracy
Kafeel et al. [24]	Vibration signals fault detection	SVM (Gaussian kernel)	98.2%
Hou et al. [25]	Energy-efficient fault detection	SVM	98%
Jiao et al. [26]	Industrial robot fault detection	Deep belief network (DBN)	97.96%
Ahn et al. [27]	Industrial robot fault detection	Conditional GAN	95.6%
Wang et al. [28]	Bearing fault detection	XGBoost	98.60%
Helmi et al. [29]	Rolling bearing fault detection	Adaptive neuro-fuzzy interface system (ANFIS)	98%
Tang et al. [30]	Turbine gearbox fault detection	LightGBM	98.67%
Hao et al. [31]	Gearbox fault detection	DBN	97%
Fawwaz et al. [32]	Hydraulic fault detection	LSTM autoencoder	98%
Chen et al. [33]	Electric Motors fault detection	Capsule Network with CNN	98%
Proposed Model	Machinery fault detection	LSTM Autoencoder	99%

When discussing objectives, the existing studies address a range of fault detection scenarios, including vibration signals, energy efficiency, and specific industrial components like robots and bearings. For instance, Kafeel et al. [24] and Hou et al. [25] targeted vibration and energy-efficient fault detection using SVM, while Ahn et al. [27] and Jiao et al. [26] focused on fault detection in industrial robots with Conditional GAN and DBN, respectively. However, our work extended the scope to general MFD, focusing on a more all-encompassing method that can handle diverse fault types.

In terms of models, a range of approaches have been utilized. SVM with Gaussian kernels [24] and other SVM variants [25] have been widespread, while models like DBN [26], Conditional GANs [27], and XGBoost [28] have shown significant promise. Our proposed model, the LSTM Autoencoder, defined a cutting-edge approach that integrated temporal sequence analysis with anomaly detection, significantly improving these methods. Notably, our model surpassed existing approaches, including DBN [31], XGBoost [28], and others, with a recorded accuracy of 99%, compared to the highest accuracy of 98.67% performed by LightGBM [30]. Regarding accuracy, our proposed LSTM Autoencoder performed a commendable 99%, outperforming the accuracy of models like SVM (98.2% to 98%), DBN (97% to 97.96%), and LightGBM (98.67%). This exceptional performance highlighted the effectiveness of our approach in capturing intricate patterns within time-series data, leading to more precise fault detection. Our model's remarkable accuracy indicated its robustness and dependability in practical scenarios, enhancing the performance of existing works.

Our work presented a significant advancement in MFD by using an LSTM Autoencoder, which differentiates itself through its ability to model sequential data effectively and provide higher accuracy. Combining sophisticated DL techniques and a comprehensive approach to fault detection sets our model apart from existing methods, promising improved performance and reliability in practical applications. This study contributes to understanding MFD and points toward possible future advances in fault detection technology. In our study, **Table 6** shows how well our model performs in comparison to earlier findings in the area of machinery failure identification. Several key measures showed how our LSTM Autoencoder model and other assessed models perform better than or in line with current methods.

V. CONCLUSION

This study explored the application of cutting-edge ML techniques for effective MFD, focusing on real-world industrial scenarios. The dataset employed in our study, sourced from Kaggle, contains 1,951 multivariate time-series samples generated by the ABVT. The MFS provides an extensive platform for assessing fault diagnosis techniques since it is intended to replicate actual machinery malfunctions in a controlled setting. This dataset is perfect for testing different fault detection algorithms in industrial settings since it focuses on three main fault types: vibration anomalies, imbalances, and alignment difficulties. The models included KNN, SVM, DNN, and an LSTM Autoencoder; each model was evaluated based on key performance metrics, with the LSTM Autoencoder performing superior results across the board, producing 99% accuracy. These findings underscore the potential of advanced DL architectures in diagnosing machinery faults with exceptional accuracy. Such predictive power is invaluable in minimizing unexpected downtime, lowering maintenance costs, and providing the longevity and reliability of critical machinery. Combining these cutting-edge MFD systems can transform maintenance procedures as industries move toward more automated and data-driven environments. Due to these models' scalability and versatility, future research should concentrate on expanding the application of this method to a broader range of manufacturing environments, improving fault diagnosis skills, and developing intelligent maintenance systems to push the boundaries.

Data Availability

No data was used to support this study.

Conflicts of Interests

The author(s) declare(s) that they have no conflicts of interest.

Funding

No funding agency is associated with this research.

Competing Interests

There are no competing interests

References

[1]. Chinedu Alex Ezeigweneme, Chinedu Nnamdi Nwasike, Adedayo Adefemi, Abimbola Oluwatoyin Adegbite, and Joachim Osheyor Gidiagba, "Smart Grids In Industrial Paradigms: A Review Of Progress, Benefits, And Maintenance Implications: Analyzing The Role Of Smart Grids In Predictive Maintenance And The Integration Of Renewable Energy Sources, Along With Their Overall Impact On The Industri," Engineering Science Technology Journal, vol. 5, no. 1, pp. 1–20, Jan. 2024, doi: 10.51594/estj.v5i1.719.

[2]. D. Neupane, M. R. Bouadjenek, R. Dazeley, and S. Aryal, "Data-Driven Machinery Fault Detection: A Comprehensive Review," 2024, doi: 10.2139/ssrn.4856168.

[3]. A. H. Sabry and U. A. B. Ungku Amirulddin, "A review on fault detection and diagnosis of industrial robots and multi-axis machines," Results in Engineering, vol. 23, p. 102397, Sep. 2024, doi: 10.1016/j.rineng.2024.102397.

[4]. S. Gawde, S. Patil, S. Kumar, P. Kamat, K. Kotecha, and A. Abraham, "Multi-fault diagnosis of Industrial Rotating Machines using Data-driven approach : A review of two decades of research," Engineering Applications of Artificial Intelligence, vol. 123, p. 106139, Aug. 2023, doi: 10.1016/j.engappai.2023.106139.

[5]. Y. LI, X. DU, F. WAN, X. WANG, and H. YU, "Rotating machinery fault diagnosis based on convolutional neural network and infrared thermal imaging," Chinese Journal of Aeronautics, vol. 33, no. 2, pp. 427–438, Feb. 2020, doi: 10.1016/j.cja.2019.08.014.

[6]. Y. Lei, J. Lin, Z. He, and M. J. Zuo, "A review on empirical mode decomposition in fault diagnosis of rotating machinery," Mechanical Systems and Signal Processing, vol. 35, no. 1–2, pp. 108–126, Feb. 2013, doi: 10.1016/j.ymssp.2012.09.015.

[7]. S. Zhang, S. Zhang, B. Wang, and T. G. Habetler, "Deep Learning Algorithms for Bearing Fault Diagnostics—A Comprehensive Review," IEEE Access, vol. 8, pp. 29857–29881, 2020, doi: 10.1109/access.2020.2972859.

[8]. D. Neupane and J. Seok, "Bearing Fault Detection and Diagnosis Using Case Western Reserve University Dataset With Deep Learning Approaches: A Review," IEEE Access, vol. 8, pp. 93155–93178, 2020, doi: 10.1109/access.2020.2990528.

[9]. D.-T. Hoang and H.-J. Kang, “A survey on Deep Learning based bearing fault diagnosis,” Neurocomputing, vol. 335, pp. 327–335, Mar. 2019, doi: 10.1016/j.neucom.2018.06.078.
[10]. S. Mushtaq, M. M. M. Islam, and M. Sohaib, “Deep Learning Aided Data-Driven Fault Diagnosis of Rotatory Machine: A Comprehensive Review,” Energies, vol. 14, no. 16, p. 5150, Aug. 2021, doi: 10.3390/en14165150.
[11]. D. Srilatha and N. Thillaiarasu, "Implementation of Intrusion detection and prevention with Deep Learning in Cloud Computing," Journal of Information Technology Management, Vol. 15, Special Issue, pp. 1-18, 2023.
[12]. B. L. R, S. Murugan, and M. Balakrishnan, “Nature Inspired Optimizers and Their Importance for AI: An Inclusive Analysis,” EAI/Springer Innovations in Communication and Computing, pp. 407–427, 2024, doi: 10.1007/978-3-031-53972-5_21.
[13]. T. Zhang et al., “Intelligent fault diagnosis of machines with small & imbalanced data: A state-of-the-art review and possible extensions,” ISA Transactions, vol. 119, pp. 152–171, Jan. 2022, doi: 10.1016/j.isatra.2021.02.042.
[14]. W. Li et al., “A perspective survey on deep transfer learning for fault diagnosis in industrial scenarios: Theories, applications and challenges,” Mechanical Systems and Signal Processing, vol. 167, p. 108487, Mar. 2022, doi: 10.1016/j.ymssp.2021.108487.
[15]. H. Zheng et al., “Cross-Domain Fault Diagnosis Using Knowledge Transfer Strategy: A Review,” IEEE Access, vol. 7, pp. 129260–129290, 2019, doi: 10.1109/access.2019.2939876.
[16]. Z. Yang, B. Xu, W. Luo, and F. Chen, “Autoencoder-based representation learning and its application in intelligent fault diagnosis: A review,” Measurement, vol. 189, p. 110460, Feb. 2022, doi: 10.1016/j.measurement.2021.110460.
[17]. J. Qian, Z. Song, Y. Yao, Z. Zhu, and X. Zhang, “A review on autoencoder based representation learning for fault detection and diagnosis in industrial processes,” Chemometrics and Intelligent Laboratory Systems, vol. 231, p. 104711, Dec. 2022, doi: 10.1016/j.chemolab.2022.104711.
[18]. M. Jalayer, C. Orsenigo, and C. Vercellis, “Fault detection and diagnosis for rotating machinery: A model based on convolutional LSTM, Fast Fourier and continuous wavelet transforms,” Computers in Industry, vol. 125, p. 103378, Feb. 2021, doi: 10.1016/j.compind.2020.103378.
[19]. J. Jiao, M. Zhao, J. Lin, and K. Liang, “A comprehensive review on convolutional neural network in machine fault diagnosis,” Neurocomputing, vol. 417, pp. 36–63, Dec. 2020, doi: 10.1016/j.neucom.2020.07.088.
[20]. M.-C. Kim, J.-H. Lee, D.-H. Wang, and I.-S. Lee, “Induction Motor Fault Diagnosis Using Support Vector Machine, Neural Networks, and Boosting Methods,” Sensors, vol. 23, no. 5, p. 2585, Feb. 2023, doi: 10.3390/s23052585.
[21]. F. An and J. Wang, “Rolling bearing fault diagnosis algorithm using overlapping group sparse-deep complex convolutional neural network,” Nonlinear Dynamics, vol. 108, no. 3, pp. 2353–2368, Mar. 2022, doi: 10.1007/s11071-022-07314-9.
[22]. X. Li, W. Zhang, and Q. Ding, “Deep learning-based remaining useful life estimation of bearings using multi-scale feature extraction,” Reliability Engineering & System Safety, vol. 182, pp. 208–218, Feb. 2019, doi: 10.1016/j.ress.2018.11.011.
[23]. M. J. Hasan, M. M. M. Islam, and J.-M. Kim, “Bearing Fault Diagnosis Using Multidomain Fusion-Based Vibration Imaging and Multitask Learning,” Sensors, vol. 22, no. 1, p. 56, Dec. 2021, doi: 10.3390/s22010056.
[24]. A. Kafeel et al., “An Expert System for Rotating Machine Fault Detection Using Vibration Signal Analysis,” Sensors, vol. 21, no. 22, p. 7587, Nov. 2021, doi: 10.3390/s21227587.
[25]. L. Hou, J. Hao, Y. Ma, and N. Bergmann, “IWSNs with On-Sensor Data Processing for Energy Efficient Machine Fault Diagnosis,” International Journal of Online and Biomedical Engineering (iJOE), vol. 15, no. 08, pp. 42–61, May 2019, doi: 10.3991/ijoe.v15i08.10314.
[26]. J. Jiao and X. Zheng, “Fault Diagnosis Method for Industrial Robots Based on DBN Joint Information Fusion Technology,” Computational Intelligence and Neuroscience, vol. 2022, pp. 1–9, Mar. 2022, doi: 10.1155/2022/4340817.
[27]. S. Ahn, J. Yoo, K.-W. Lee, B. D. Youn, and S.-H. Ahn, “Frequency-focused sound data generator for fault diagnosis in industrial robots,” Journal of Computational Design and Engineering, vol. 11, no. 4, pp. 234–248, Jul. 2024, doi: 10.1093/jcde/qwae061.
[28]. H. Wang et al., “An improved bearing fault detection strategy based on artificial bee colony algorithm,” CAAI Transactions on Intelligence Technology, vol. 7, no. 4, pp. 570–581, Jun. 2022, doi: 10.1049/cit2.12105.
[29]. H. Helmi and A. Forouzantabar, “Rolling bearing fault detection of electric motor using time domain and frequency domain features extraction and ANFIS,” IET Electric Power Applications, vol. 13, no. 5, pp. 662–669, Feb. 2019, doi: 10.1049/iet-epa.2018.5274.
[30]. M. Tang et al., “An Improved LightGBM Algorithm for Online Fault Detection of Wind Turbine Gearboxes,” Energies, vol. 13, no. 4, p. 807, Feb. 2020, doi: 10.3390/en13040807.
[31]. H. Hao et al., “Gear Fault Detection in a Planetary Gearbox Using Deep Belief Network,” Mathematical Problems in Engineering, vol. 2022, pp. 1–12, Mar. 2022, doi: 10.1155/2022/9908074.
[32]. D. Z. Fawwaz and S.-H. Chung, “Real-Time and Robust Hydraulic System Fault Detection via Edge Computing,” Applied Sciences, vol. 10, no. 17, p. 5933, Aug. 2020, doi: 10.3390/app10175933.
[33]. J. Chen et al., “Novel Data-Driven Approach Based on Capsule Network for Intelligent Multi-Fault Detection in Electric Motors,” IEEE Transactions on Energy Conversion, vol. 36, no. 3, pp. 2173–2184, Sep. 2021, doi: 10.1109/tec.2020.3046642.