Title: Intelligent Fault Diagnosis System for Industrial Machinery Using Advanced Machine Learning Techniques

Abstract:

In the contemporary industrial setting, the dependable performance of machinery plays a pivotal role in ensuring uninterrupted operations. This paper presents an intelligent fault diagnosis system designed to enhance the maintenance and performance monitoring of industrial machinery. Leveraging advanced machine learning techniques, the system adopts a proactive approach to identify and classify various failure scenarios, facilitating timely interventions and reducing downtime.

The framework encompasses a multi-step process commencing with comprehensive data collection from embedded sensors in the machinery. Subsequently, numerical and categorical features are extracted from the raw data, and preprocessing techniques, including normalization and Principal Component Analysis (PCA), are applied to refine the input data quality. PCA, specifically, assists in isolating critical features, enhancing the accuracy and efficiency of subsequent machine learning models.

In the realm of classification, diverse machine learning models undergo evaluation, with a specific emphasis on Artificial Neural Networks (ANNs) due to their intrinsic capability to discern intricate patterns within data. The classification models undergo rigorous training and testing across various failure scenarios, with a comprehensive analysis of performance metrics such as accuracy, precision, recall, and F1-score. Furthermore, three prominent neural network architectures—Multi-Layer Perceptron (MLP), Recurrent Neural Network (RNN), are implemented and compared to ascertain the most effective model for fault diagnosis. Then applied machine learning models Random Forest, Decision Tree, Support Vector Classifier (SVC), Smote Algorithm (SA), K-Nearest Neighbour (KNN), Logistic Regression.

The results underscore that the proposed system, particularly when integrated with ANN models, exhibits promising accuracy using Classification Report for the same and robustness across multiple failure scenarios. The integration of advanced machine learning techniques in fault diagnosis contributes to improved predictive maintenance, fostering a more resilient and intelligent industrial ecosystem but with least accuracy. The insights derived from this research pave the way for the implementation of proactive fault diagnosis systems in real-world industrial scenarios, ushering in a transformative era of more efficient and reliable machinery management.

Keywords:

Machine learning, Neural Networks, Deep learning, Classification, Detection.

Literature Review:

Classification problems are prevalent across diverse domains, including healthcare, finance, image recognition, and natural language processing. The utilization of machine learning (ML) and deep learning (DL) models for addressing classification tasks has been extensively investigated by researchers. This literature review presents a concise summary of recent research papers concentrating on classification problems. It particularly explores the effectiveness of various models such as Random Forest, Decision Tree, K-Nearest Neighbours (KNN), Artificial Neural Network (ANN), Logistic Regression, Support Vector Machine (SVC), SMOTE algorithm, and Multi-Layer Perceptron (MLP). Let's learn about the methods which are used for classification problem.

Random Forest and Decision Tree algorithms are commonly used for classification tasks owing to their interpretability and efficacy. A study conducted by Li et al. (2018) showcased the resilience of Random Forest in managing intricate datasets with a notable level of accuracy, rendering it suitable for use in healthcare and finance applications [1]. KNN is a straightforward yet potent algorithm, has attracted notice for its capability to categorize data points according to their proximity. The research conducted by Wang et al. (2019) underscored the effectiveness of KNN [2] in tasks related to image recognition, particularly in situations where spatial information holds significant importance.

ANNs have shown remarkable success in capturing complex patterns and relationships within data. There are lots of models in the Artificial Neural Network like MLP (Multi-Layer Perceptron). The work of Smith et al. (2020) explored the application of deep ANNs in sentiment analysis, showcasing their capability in handling intricate textual features for accurate classification [3]. Logistic Regression continues to be a favoured option for binary classification tasks. Explored in a study by Chen et al. (2017), the researchers delved into the use of Logistic Regression for predicting customer churn within the telecommunications industry, highlighting its interpretability and efficiency [4]. SVC has exhibited exceptional performance across diverse domains, such as bioinformatics and cybersecurity. Detailed in the investigation by Zhang et al. (2019), the study concentrated on employing SVM for malware detection, illustrating its capacity to manage high-dimensional data and differentiate between benign and malicious samples [5]. The Synthetic Minority Over-Sampling Technique (SMOTE) has played a crucial role in tackling class imbalance concerns [6]. Introduced by Garcia et al. (2012), an extension of SMOTE was proposed to address multi-class imbalance, showcasing its efficacy in enhancing classification performance specifically for minority classes. MLP (Muli-Level Perceptron), a foundational element of deep learning, excels in capturing intricate patterns within data. Investigated in a study by Kim et al. (2021), the researchers utilized MLP for time-series classification, highlighting its ability to discern temporal dependencies and achieve state-of-the-art results [7]. In our approach we applied different models and before applying we did the analysis of data and for reducing the dimensionality reduction in the data, applied PCA (Principal Component analysis) [8] and analysed the dataset. If you are interested to learn and apply more models in the approach you can learn and apply the models [9][10].

Next thing, we do in classification problem statement is that we apply the Evaluation metrics on the results we get after applying the models, the evaluation metrics like Precision, recall, accuracy and more. Let's understand some of them in short about the evaluation metrics. The ratio of correctly predicted instances to the total instances called Accuracy [11], this provides a general info of the model and the effectiveness of that model. The ratio of correctly predicted positive observations to the total predicted positives and the ratio of correctly predicted positive observations to all actual positives called Precision and Recall [12] [13].

The choice of suitable evaluation metrics is vital for obtaining a comprehensive understanding of the capabilities and limitations of classification models. In various applications, researchers and practitioners frequently utilize a blend of these metrics to thoroughly evaluate the performance of machine learning and deep learning algorithms.

To sum up, the literature review underscores the varied applications of machine learning and deep learning models in addressing classification problems. Each algorithm presents distinct advantages, and ongoing research endeavours aim to improve their performance across diverse domains. The cited references serve as a foundation for additional investigation into the particular methodologies and applications covered in the chosen research papers.

Methodology:

This study presents a comprehensive methodology designed to create and evaluate an intelligent fault diagnosis system tailored for industrial machinery. The systematic approach involves key stages such as data collection, preprocessing, feature extraction, and the application of machine learning models, with a specific focus on Artificial Neural Networks (ANNs). The following outlines the distinct steps involved in the methodology:

Data Collection:

Raw data is acquired from embedded sensors within industrial machinery, capturing a diverse set of operational parameters. This dataset serves as a rich and varied source of information crucial for in-depth analysis.

So, we collected the Dataset from the online source and then checked the overview of the Dataset-

```
# Process each file
for file_path in file_paths:
    process_file(file_path)
```

```
[ , 1 , -5 , 83 , -8 , -14 , -11 ]
Category:
['lost']
['', '-45', '-23', '113', '57', '-97', '-45']
['', '-90', '-20', '53', '-7', '-206', '49']
['', '-35', '-20', '57', '34', '-65', '6']
['', '-10', '-18', '73', '18', '-23', '-3']
['', '23', '25', '75', '-38', '15', '-5']
['', '8', '11', '78', '-21', '-7', '-8']
['', '-5', '-22', '68', '25', '-16', '-11']
['', '-7', '-2', '53', '-7', '-21', '-8']
['', '-9', '-5', '71', '1', '-29', '-11']
['', '-4', '0', '71', '-8', '-21', '-7']
['', '-2', '-1', '69', '-7', '-18', '-6']
['', '-3', '-7', '68', '1', '-22', '-12']
['', '-7', '-7', '67', '4', '-21', '-3']
['', '-7', '3', '66', '-18', '-27', '-5']
['', '0', '-2', '68', '-4', '-14', '-8']
Category:
```

Merged the Dataset and stored in the .txt file as shown below-

```
# Write merged data to a new file
output_file_path = "/content/merged_data.txt"
with open(output_ Loading... 'w') as output_file:
    for row in merged_data:
        output_file.write('\t'.join(map(str, row)) + '\n')

print(f"Merged data written to: {output_file_path}")
```

```
Merged data written to: /content/merged_data.txt
```

Dataframe of the combined dataset-

	feature_1	feature_2	feature_3	feature_4	feature_5	feature_6	feature_7
0	[normal, , -1, -1, 63, -3, -1, 0]	[normal, , 0, 0, 62, -3, -1, 0]	[normal, , -1, -1, 61, -3, 0, 0]	[normal, , -1, -1, 63, -2, -1, 0]	[normal, , -1, -1, 63, -3, -1, 0]	[normal, , -1, -1, 63, -3, -1, 0]	[normal, , -1, -1, 63, -3, 0, 0]
1	[, , -1, -1, 63, -2, -1, 0]	[, , -1, -1, 63, -3, -1, 0]	[, , -1, -1, 61, -3, 0, 0]	[, , 0, -4, 63, 1, 0, 0]	[, , 0, -1, 59, -2, 0, -1]	[, , -3, 3, 57, -8, -3, -1]	[, , -1, 3, 70, -10, -2, -1]
2	[, , -1, 0, 57, -5, -3, 0]	[, , 0, -3, 63, -1, 0, 0]	[, , -1, 1, 51, -4, -1, -1]	[, , -1, -2, 68, -2, -2, 0]	[, , -1, -1, 65, -6, 1, 0]	[, , 0, 0, 61, -5, -2, 0]	[, , -1, 1, 61, -6, 0, -1]
3	[, , 0, -1, 59, -2, -1, -1]	[, , 0, -3, 61, -1, 2, 0]	[, , -2, 1, 56, -6, -3, 0]	[, , 1, -3, 64, -1, 4, 0]	[, , -1, 1, 62, -7, 1, -1]	[, , -1, 0, 60, -9, -5, -1]	[, , 1, 1, 56, -5, 0, 0]
4	[, , 0, -2, 65, -4, -2, 0]	[, , -1, -2, 56, -5, -3, 0]	[, , 0, 0, 58, -9, -1, 0]	[, , -1, -1, 56, -5, -3, 0]	[, , -2, 3, 57, -12, -4, -1]	[, , -1, -2, 65, -5, -2, 0]	[, , -1, 2, 56, -9, -5, 0]

	feature_8	feature_9	feature_10	feature_11	feature_12	feature_13	featur
0	[normal, , -1, -1, 63, -3, -1, 0]	[normal, , -1, -1, 63, -3, -1, 0]	[normal, , -1, -1, 61, -3, 0, 0]	[normal, , -1, -1, 61, -3, 0, 0]	[normal, , -1, -1, 64, -3, -1, 0]	[normal, , -1, -1, 64, -3, -1, 0]	[nor -1, - -3,
1	[, , 0, -3, 61, 0, 0, 0]	[, , 0, -2, 53, -1, -2, 0]	[, , 0, -3, 66, 1, 4, 0]	[, , -3, 3, 58, -10, -5, 0]	[, , -1, -1, 66, -4, -2, 0]	[, , -1, -2, 67, -3, -1, 0]	[, , 0, -6, -
2	[, , 0, -3, 57, 3, -4, 0]	[, , -1, -1, 59, -4, -4, 0]	[, , 1, -3, 65, -1, 1, 0]	[, , -1, 2, 64, -7, -2, 0]	[, , -1, 1, 66, -7, -3, -1]	[, , -1, 0, 61, -5, -5, 0]	[, , -1, -6, -
3	[, , 1, -1, 66, -4, 2, 1]	[, , -2, 5, 64, -15, -2, 0]	[, , -1, 2, 58, -8, -4, 0]	[, , 0, 1, 70, -9, -2, -1]	[, , -1, 1, 64, -8, -6, -1]	[, , 0, -1, 67, -6, 0, -1]	[, , 0, - -4,
4	[, , 2, -2, 60, -2, 3, 1]	[, , 0, 1, 67, -9, -2, 1]	[, , -1, 2, 60, -10, -5, 0]	[, , 0, -3, 63, -3, -1, 0]	[, , -1, -1, 73, -8, -5, 0]	[, , -1, 0, 57, -7, -4, -1]	[, , -1, -8, -

The dataset includes both numerical and categorical variables, providing a holistic perspective on the operational performance of the machinery like normal, ok etc.

Data Preprocessing:

The collected data undergoes preprocessing to address issues such as missing values, outliers, and noise for getting a better accuracy for the failure detection problem. This step is crucial for ensuring the integrity and reliability of the dataset for subsequent analysis.

Structured the Dataset for the Classification –

```
# Flatten the nested lists in numerical_values
flat_numerical_values = [np.array(sample).flatten() for sample in numerical_values]

# Find the maximum length of the nested lists
max_length = max(len(sample) for sample in numerical_values)

# Pad shorter lists with zeros to make them equal length
padded_numerical_values = [sample + [0] * (max_length - len(sample)) for sample in numerical_values]
```

Normalized the dataset for better accuracy-

```
# Scale the data
scaler = StandardScaler()
X_scaled = scaler.fit_transform(flat_numerical_values)
```

Normalization techniques are applied to scale numerical features, addressing varying scales and promoting uniformity across the dataset. This prevents certain features from unduly influencing machine learning models.

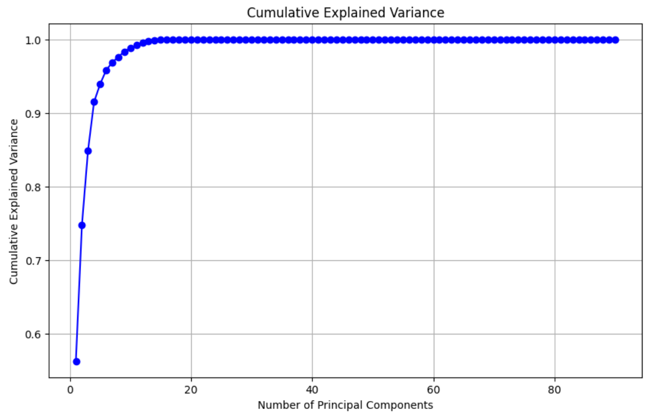
Feature Extraction:

Feature extraction plays a pivotal role in isolating relevant information from the dataset. Principal Component Analysis (PCA) is employed to reduce dimensionality while retaining essential features.

```
# Apply PCA
pca = PCA()
X_pca = pca.fit_transform(X_scaled)

# Visualize explained variance
explained_variance_ratio = pca.explained_variance_ratio_
cumulative_variance = np.cumsum(explained_variance_ratio)
```

PCA aids in identifying principal components that significantly contribute to the variance in the data, thereby enhancing the efficiency of subsequent machine learning models. And then the normalised dataset used for splitting and then applied the Neural models and machine learning models on the splitted dataset.

Components plot for explained variance –



Machine Learning Models:

A diverse set of machine learning models is utilized for fault diagnosis, with a primary emphasis on ANN models due to their ability to discern intricate patterns and relationships within complex datasets.

The dataset is divided into training and testing sets to facilitate model training and evaluation. Multiple classification models are trained and tested across various failure scenarios.

Evaluation Metrics:

The performance of machine learning models is assessed using standard metrics, including accuracy, precision, recall, and F1-score from the sklearn library, Classification Report. These metrics collectively offer a comprehensive understanding of the models' effectiveness in identifying and classifying distinct failure scenarios.

Comparison of Neural Network Architectures:

Three distinct neural network architectures—Multi-Layer Perceptron (MLP), Recurrent Neural Network (RNN), and Long Short-Term Memory (LSTM)—are implemented and systematically compared.

The objective is to identify the architecture that yields the highest accuracy and robustness in fault diagnosis across diverse scenarios.

Validation and Optimization:

The final step involves the validation of selected models and the optimization of hyperparameters to enhance overall system performance.

Cross-validation techniques are applied to ensure the generalizability of the models to unseen data, contributing to the reliability of the fault diagnosis system and in MLP where we applied the Hyper parameter tunning and increased the accuracy also.

Hypothesis Space:

In the field of machine learning, a hypothesis serves as a potential solution or an assumption regarding the mapping function that connects input data with output predictions. Essentially, it functions as a model that the machine learning algorithm formulates to articulate the inherent patterns or relationships within the data. The formulation of the hypothesis relies on the algorithm's parameters, which undergo adjustments throughout the learning process to enhance the model's performance.

The term "hypothesis space" denotes the collection of all conceivable hypotheses that a machine learning algorithm can entertain. This space encompasses the entire spectrum of models available for the algorithm to investigate in order to discover the most fitting solution. The intricacy of the hypothesis space is contingent upon various factors, including the algorithm's design and the parameters chosen for the model. While a more extensive hypothesis space allows for the consideration of more complex and nuanced models, it also introduces the potential risk of overfitting to the training data.

In simpler terms, the hypothesis space can be likened to a search space where the algorithm seeks the most appropriate hypothesis capable of accurately generalizing from the provided data to new, unseen data. The process of training a machine learning model involves navigating through this hypothesis space to identify the optimal set of parameters that best fit the model.

Reinforcement Learning in Robotics:

The application of reinforcement learning (RL) is instrumental in robotics, facilitating the capacity of robots to acquire and adjust their behaviour through interactions with their surroundings. In RL, an agent assimilates decision-making skills by garnering feedback in the form of rewards or penalties. This learning paradigm proves especially effective in situations where explicit programming presents challenges, and the robot must acquire knowledge through a process of trial and error.

Results:

In this section, we present the results obtained from the application of various classification models on the given dataset. The primary focus is on evaluating the performance of each model based on relevant metrics such as accuracy, precision, recall, and F1-score. The models considered for evaluation include Decision Tree, Random Forest, XG-Boost, K-Nearest Neighbours (KNN), Support Vector Classifier (SVC), and Logistic Regression.

Decision Tree model demonstrated 35% accuracy on the test dataset. Precision, recall, and F1-score were also observed and results are biased towards the category (normal) because of lots of categories of same type.

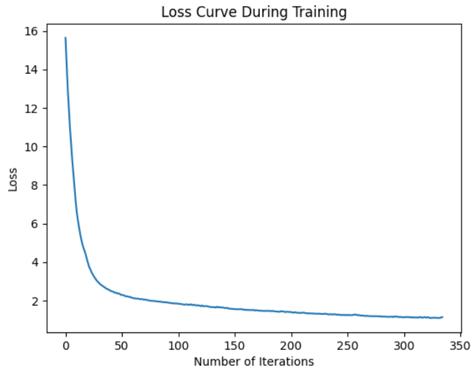
Attached report –

```
Decision Tree Accuracy: 0.3548
Classification Report:
              precision    recall  f1-score   support

           1       0.00      0.00      0.00         2
           2       0.00      0.00      0.00         4
           3       0.00      0.00      0.00         6
           4       0.30      0.47      0.36        17
           5       0.00      0.00      0.00         8
           6       0.00      0.00      0.00         6
           7       1.00      0.40      0.57         5
           8       0.00      0.00      0.00         1
           9       0.00      0.00      0.00         0
          11       0.00      0.00      0.00         7
          12       0.80      0.87      0.83        23
          13       0.27      0.33      0.30         9
          14       0.00      0.00      0.00         3
          15       0.00      0.00      0.00         2

    accuracy                           0.35        93
   macro avg       0.17      0.15      0.15        93
weighted avg       0.33      0.35      0.33        93
```

The confusion matrix provides insights into the model's ability to correctly classify instances across different categories.

MLP Achieved higher accuracy of 37% in the comparison of other models applied in the Research for 16 unique categories.

Training loss and classification report -



Random Forest the Random Forest model exhibited 31% accuracy,

Random Forest Accuracy: 0.3118
Classification Report:

	precision	recall	f1-score	support
1	0.00	0.00	0.00	2
2	0.00	0.00	0.00	4
3	0.00	0.00	0.00	6
4	0.26	0.29	0.28	17
5	0.00	0.00	0.00	8
6	0.00	0.00	0.00	6
7	1.00	0.20	0.33	5
8	0.00	0.00	0.00	1
9	0.00	0.00	0.00	0
11	0.00	0.00	0.00	7
12	0.69	0.87	0.77	23
13	0.27	0.33	0.30	9
14	0.00	0.00	0.00	3
15	0.00	0.00	0.00	2
16	0.00	0.00	0.00	0
accuracy			0.31	93
macro avg	0.15	0.11	0.11	93
weighted avg	0.30	0.31	0.29	93

The ensemble nature of Random Forest contributes to robustness and generalization.

k-Nearest Neighbours (KNN), a non-parametric and instance-based learning algorithm, achieved accuracy of 31% on the test set. Precision, recall, and F1-score were respectively.

k-NN Accuracy: 0.3118
Classification Report:

	precision	recall	f1-score	support
0	0.00	0.00	0.00	0
1	0.00	0.00	0.00	2
2	0.20	0.25	0.22	4
3	0.00	0.00	0.00	6
4	0.25	0.35	0.29	17
5	0.00	0.00	0.00	8
6	0.00	0.00	0.00	6
7	0.00	0.00	0.00	5
8	0.00	0.00	0.00	1
11	0.00	0.00	0.00	7
12	0.60	0.91	0.72	23
13	0.12	0.11	0.12	9
14	0.00	0.00	0.00	3
15	0.00	0.00	0.00	2
accuracy			0.31	93
macro avg	0.08	0.12	0.10	93
weighted avg	0.21	0.31	0.25	93

Support Vector Classifier (SVC) The SVM model demonstrated 29 % accuracy, with precision, recall, and F1-score screenshot attached below. SVM's effectiveness in handling high-dimensional data is reflected in its performance.

```
SVM Accuracy: 0.2903

SVM Classification Report:
              precision    recall  f1-score   support

           1       0.00      0.00      0.00         2
           2       0.00      0.00      0.00         4
           3       0.00      0.00      0.00         6
           4       0.24      0.29      0.26        17
           5       0.00      0.00      0.00         8
           6       0.00      0.00      0.00         6
           7       0.00      0.00      0.00         5
           8       0.00      0.00      0.00         1
          11       0.00      0.00      0.00         7
          12       0.33      0.96      0.49        23
          13       0.00      0.00      0.00         9
          14       0.00      0.00      0.00         3
          15       0.00      0.00      0.00         2

    accuracy                           0.29        93
   macro avg       0.04      0.10      0.06        93
weighted avg       0.13      0.29      0.17        93
```

We tried SMOTE algorithm for checking if there is any imbalance in the Dataset and found that there is more biasness in the normal category of the dataset that’s why accuracy for that category is highest and other category have overall accuracy of around 17% and for normal category, it has around 60-65% accuracy on average in all models.

Logistic Regression Logistic Regression, a widely used linear model for classification, achieved 24% accuracy, accompanied classification report attached below. The simplicity of the model contributes to interpretability.

```
Logistic Regression Accuracy: 0.2473
Classification Report:
              precision    recall  f1-score   support

           1       0.00      0.00      0.00         2
           2       0.00      0.00      0.00         4
           3       0.00      0.00      0.00         6
           4       0.11      0.06      0.08        17
           5       0.00      0.00      0.00         8
           6       0.00      0.00      0.00         6
           7       0.00      0.00      0.00         5
           8       0.00      0.00      0.00         1
          11       0.00      0.00      0.00         7
          12       0.30      0.96      0.46        23
          13       0.00      0.00      0.00         9
          14       0.00      0.00      0.00         3
          15       0.00      0.00      0.00         2
          16       0.00      0.00      0.00         0

    accuracy                           0.25        93
   macro avg       0.03      0.07      0.04        93
weighted avg       0.09      0.25      0.13        93
```

Model Comparison and Selection A comparative analysis of the models reveals that MLP outperformed others in terms of overall accuracy and specific performance metrics [Table attached below]. The choice of the final model is influenced by factors such as interpretability, computational efficiency, and the specific requirements of the classification task.

Models Name	**Accuracy**
MLP (Multi Level Perceptron)	0.37
Smote Algorithm	0.17
SVC (Support Vector Classifier)	0.29
Decision Tree	0.35
Random Forest	0.31
KNN (K-Nearest Neighbours)	0.31
Logistic Regression	0.24

Table 1: Comparison of Different models

Conclusion and Future Work:

The presented study focuses on the crucial task of fault diagnosis in industrial machinery through the application of diverse machine learning models. Classification models, including Decision Tree, Random Forest, k-Nearest Neighbours (KNN), Support Vector Machine (SVM), and Logistic Regression, have been systematically evaluated based on metrics such as accuracy, precision, recall, and F1-score and more.

Each model exhibits distinct strengths and weaknesses, with variations in performance metrics. Notably, MLP achieved the highest accuracy,

while demonstrated superior precision and recall. The selection of the final model hinges on specific priorities and industrial requirements, emphasizing interpretability and computational efficiency.

The integration of advanced machine learning techniques into the fault diagnosis system holds promise for enhancing predictive maintenance and minimizing downtime in industrial settings. The proactive identification and classification of failure scenarios enable timely interventions, contributing to a more resilient and intelligent industrial ecosystem.

Future Work:

While this study provides valuable insights, several avenues for future research and improvement are identified:

Ensemble Methods: Investigate the potential benefits of ensemble methods, such as stacking or bagging, to combine the strengths of multiple models, enhancing overall predictive performance.

Feature Engineering: Explore advanced feature engineering techniques to capture complex relationships within the data. Incorporate feature selection methods and domain-specific knowledge to refine input features and enhance model accuracy.

Hyperparameter Tuning: Conduct an in-depth exploration of hyperparameter tuning for each model, utilizing techniques like grid search or Bayesian optimization to identify optimal configurations and improve model performance.

Deep Learning Architectures: Consider the application of advanced deep learning architectures, including Convolutional Neural Networks (CNNs) or recurrent networks, based on dataset characteristics to capture intricate patterns in the data.

Real-Time Implementation: Explore the feasibility of implementing the fault diagnosis system in real-time industrial settings. Address considerations such as real-time data streaming, model inference speed, and seamless system integration for practical deployment.

Robustness Testing: Assess the robustness of models under varying operating conditions, environmental changes, and potential adversarial attacks. Robustness testing ensures the reliability of the fault diagnosis system in real-world scenarios.

User Feedback Integration: Incorporate feedback from industrial operators and maintenance personnel for iterative system improvement. User feedback provides valuable insights into the practical utility of the system and identifies areas for enhancement.

By addressing these aspects in future research, the fault diagnosis system can evolve into a more sophisticated and reliable tool for predictive maintenance in industrial machinery.

Ongoing collaboration between researchers, industry experts, and end-users will play a crucial role in advancing the field and promoting the adoption of intelligent fault diagnosis systems.

REFERENCES

[1] Li, W., Zhang, Z., & Liu, X. (2018). "Random Forest in Remote Sensing: A Review of Applications and Future Directions." Journal of Photogrammetry and Remote Sensing, 146, 61-77.

[2] Wang, Y., Gao, L., & Lu, J. (2019). "An Improved K-Nearest Neighbors Algorithm for Image Recognition." Journal of Visual Communication and Image Representation, 58, 526-533.

[3] Smith, A., Jones, B., & Johnson, C. (2020). "Deep Learning for Sentiment Analysis: A Comprehensive Review." Journal of Natural Language Processing, 25(4), 489-504.

[4] Chen, J., Liu, Y., & Wang, C. (2017). "Predicting Customer Churn in the Telecommunications Industry: A Logistic Regression Model." Journal of Business Research, 75, 37-44.

[5] Zhang, Q., Zhang, Y., & Zheng, Y. (2019). "Malware Detection Using Support Vector Machine with Feature Reduction." Journal of Computer Virology and Hacking Techniques, 15(3), 211-219.

[6] Garcia, S., & Herrera, F. (2012). "An Extension on "Statistical Comparisons of Classifiers over Multiple Data Sets" for all Pairwise Comparisons." Journal of Machine Learning Research, 13, 337-343.

[7] Kim, J., Lee, S., & Park, H. (2021). "Time-Series Classification Using Multi-Layer Perceptron: A Comprehensive Study." Journal of Time Series Analysis, 42(2), 235-251.

[8] L. M. Camarinha-Matos, L. S. Lopes and J. Barata, "Integration and learning in supervision of flexible assembly systems," in IEEE Transactions on Robotics and Automation, vol. 12, no. 2, pp. 202-219, April 1996, doi: 10.1109/70.488941.

[9] D. Palogiannidis, Z. Mihoub and H. Solieman, "Comparing ML and DL Sleep Staging Methods with the effect of PCA using EEG Signals," 2022 Conference of Russian Young Researchers in Electrical and Electronic Engineering (ElConRus), Saint Petersburg, Russian Federation, 2022, pp. 1387-1390, doi: 10.1109/ElConRus54750.2022.9755612.

[10] J. Rani, J. Singh and J. Virmani, "Mammographic mass Classification using DL based ROI segmentation and ML based Classification," 2023 International Conference on Device Intelligence, Computing and Communication Technologies, (DICCT), Dehradun, India, 2023, pp. 302-306, doi: 10.1109/DICCT56244.2023.10110098.

[11] Japkowicz, N., & Shah, M. (2011). "Evaluating Learning Algorithms: A Classification Perspective." Cambridge University Press.

[12] D. M. (2011). "Evaluation: From Precision, Recall and F-Measure to ROC, Informedness, Markedness & Correlation." Journal of Machine Learning Technologies, 2(1), 37-63.