Journal, Advances in Mathematical & Computational Sciences
Vol. 12 No. 4, 2024 Series
www.isteams.net/mathematics-computationaljournal

Journal of Advances in Mathematical & Computational Sciences
An International Pan-African Multidisciplinary Journal of the SMART Research Group
International Centre for IT & Development (ICITD) USA
© Creative Research Publishers
Available online at https://www.isteams.net/ mathematics-computationaljournal.info
CrossREF Member Listing - https://www.crossref.org/06members/50go-live.html

Development of A Predictive Model For Equipment Failure And Maintenance Scheduling Using An Optimised Support Vector Machine

*[1]Adoyi, O. P., [2]Adeosun, O. O., [3]Olabiyisi, S. O., [4]Adeniran, J. O., [5]Fagbote, O. O. & [6]Olajide B. O.
[1,2,3] Department of Computer Science, Ladoke Akintola University of Technology, Oyo State. Nigeria
[4] Department of Computer Science, Osun State University, Osogbo, Osun State. Nigeria
[5] Agriculture Develop & Management Department, Agric & Rural Management Training Institute (ARMTI).
[6]Department of Computer Science, University of Ilesh, Osun State. Nigeria
*corresponding author: adoyipatience710@gmail.com

ABSTRACT

Equipment failure in industries causes downtime, productivity losses, and high maintenance costs. Traditional reactive and preventive maintenance approaches are limited by either responding too late or following fixed schedules, leading to unnecessary or missed interventions. Predictive models also struggle with class imbalance, as failure events are rare, causing biased predictions. This research proposed an optimised Support Vector Machine (SVM) model to address these issues, enhancing prediction accuracy and reducing downtime by identifying failures early. The research data was obtained from the University of California Irvine Machine Learning Repository. The research methodology involves a combination of techniques to enhance the SVM model's performance. First, dataset class imbalance was addressed using Synthetic Minority Over-Sampling Technique (SMOTE) to generate synthetic failure events and random under-sampling to balance the dataset. Then, hyperparameter tuning via grid search optimised the SVM's regularisation and kernel parameters. Finally, the Stochastic Gradient Descent (SGD) optimisation algorithm was applied to improve the SVM predictive model. The results of the optimised SVM model showed an accuracy of 95%, with a recall of 0.80 and an AUC score of 0.96, indicating good predictive capabilities. While the precision was relatively low, the high recall ensured that most failures were correctly identified, which is essential for predictive maintenance applications. The model demonstrated its ability to balance trade-offs between recall and precision, ultimately providing a better prediction tool for equipment failure.

Keywords: Predictive Model, Equipment Failure, Optimisation, SVM, SGD.

Adoyi, O. P., Adeosun, O. O., Olabiyisi, S. O., Adeniran, J. O., Fagbote, O. O. & Olajide B. O. (2024): Development of A Predictive Model For Equipment Failure And Maintenance Scheduling Using An Optimised Support Vector Machine. Journal of Advances in Mathematical & Computational Science. Vol. 12, No. 3. Pp 45-56. Available online at www.isteams.net/mathematics-computationaljournal. dx.doi.org/10.22624/AIMS/MATHS/V12N4P4

1. INTRODUCTION

Equipment failures are often caused by wear and tear, which results from repeated use and exposure to various environmental factors such as temperature, humidity, and vibration (Goswami and Rai, 2023). Over time, this leads to degradation of components and eventual failure. However, the rate at which these processes occur varies depending on several factors, making it challenging to predict precisely when a failure will occur. According to Sezer et al. (2018), maintenance costs constitute between 15% and 40% of the total costs for producing goods. Notably, the primary source of financial strain attributed to non-fixed maintenance activities, encompassing interventions targeted at addressing sudden breakdowns of machinery and production-related blockages. Analysis of the maintenance costs underscores a critical disparity: interventions carried out unexpectedly incur a cost approximately three times higher than those programmed in advance (Ren, 2021; Wang and Gao, 2022). Hence, it is unsurprising that, with the onset of Industry 4.0 heralding the anticipated fourth industrial revolution, the maintenance sector stands out as a focal point for substantial investments and heightened research activity (Zhong et al., 2017).

According to Leukel et al. (2021), Machine learning is significant in transitioning from reactive to predictive maintenance strategies, emphasising the need for predictive models to forecast equipment failures and facilitate strategic scheduling of maintenance activities. Similarly, Lee et al. (2020) and Sezer et al. (2018) submitted that elucidating these cost dynamics underscores industries' need to adopt predictive maintenance approaches to optimise operational expenditures and bolster overall efficiency in production processes. Predictive maintenance emerges as a viable solution when implementing this system for real-world scenarios (Leukel et al., 2021). Leveraging advanced analytics and machine learning algorithms, predictive maintenance models can filter out noise in sensor-collected data, optimise computational costs through efficient algorithms, and utilise historical data to predict equipment health states (Kotsiopoulos et al., 2020). Also, predictive maintenance reduces the dependence on human certification, offering a more streamlined and accurate approach to maintenance scheduling and intervention.

Different works of literature have reported on the predictive model for equipment failure. Campos et al. (2020) and Mohammed et al. (2019) explored the use of machine learning (ML) for failure prediction, showing that different algorithms performed variably depending on the context. While Ren (2021) emphasised the need for careful algorithm selection, (Wang and Gao, 2022) submitted that support vector machines (SVM) achieved a 90% prediction accuracy, outperforming other models. Also, Celikmih et al. (2020) introduced a hybrid model using feature selection and data elimination techniques, showing its effectiveness in predicting aircraft system failures. Jimenez et al. (2020) focused on predictive maintenance in the shipping sector, highlighting the potential of computational intelligence for real-time equipment monitoring. However, these studies highlight a need for a comprehensive framework for optimising machine learning algorithm selection for predictive maintenance. While Luis et al. (2021) demonstrated the potential of AutoML for such tasks, they did not address the manual fine-tuning of individual algorithms. Shaheen et al. (2023) showed the value of feature selection techniques but did not probe into algorithm optimisation. This research aims to bridge this gap by developing an optimised SVM model for predicting equipment failure, improving prediction performance and maintenance scheduling.

2. METHODOLOGY

The research employed a systematic approach to develop a predictive model for equipment failure using historical data. It began with data collection, followed by data preprocessing and the implementation of an SVM algorithm to detect failure patterns. The model was optimised by fine-tuning hyperparameters (regularisation parameter and kernel function) through techniques like grid search and cross-validation. This optimisation aimed to enhance the model's accuracy and generalizability. Finally, the model's performance was assessed using precision, recall, and F1 score to ensure reliability.

2.1 Model Architecture

The failure prediction model follows a three-phase approach: pre-processing, training, and prediction. Historical failure data and performance indicators are curated, cleaned, and feature-engineered to improve input quality. An exploratory analysis identifies patterns, guiding the selection of the Support Vector Machine (SVM) algorithm for prediction.

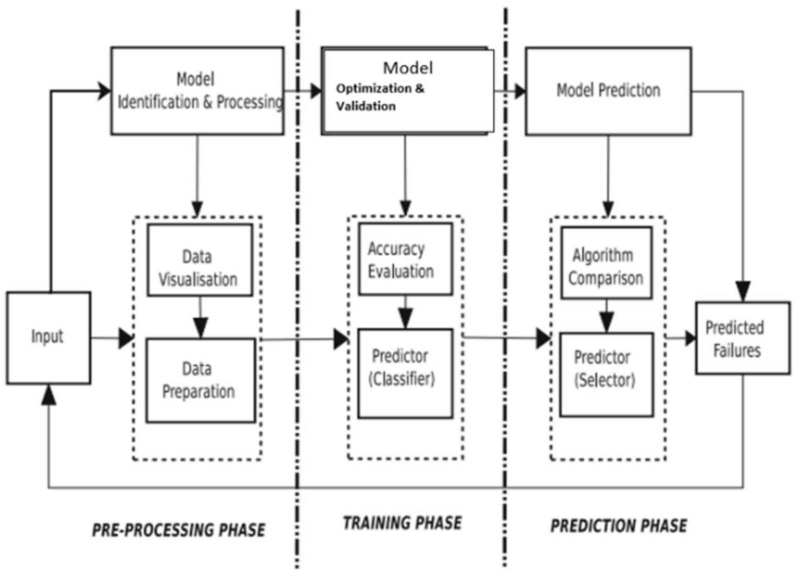

Figure 1: The model architecture

2.2 Data Collection and Pre-Processing

Obtaining real-world industrial predictive maintenance datasets can be challenging due to confidentiality and proprietary concerns. As a solution, researchers often use publicly accessible repositories with simulated datasets, like the University of California Irvine Machine Learning Repository, which provided the dataset for this research. The dataset as shown in Table 1 contains 10,000 data points across 14 features, capturing key variables influencing machine performance. These include product ID, air and process temperatures, rotational speed, applied torque, tool wear, and binary "failure" labels indicating operational success or failure. The dataset consists of input and output variables representing system functionality, as outlined in Table 1. The data was normalised to standardise numerical variables, preventing dominant features from skewing machine learning (ML) algorithms. Feature engineering was employed to create or modify variables, enhancing the dataset's ability to capture relevant patterns. The dataset was then split into training and testing sets, with K-fold cross-validation used to improve model robustness and prevent overfitting during training. Finally, the processed data was used for ML model optimisation.

Table 1: Equipment Failure Dataset Description

UDI	Air temperature [K]	Process temperature [K]	Rotational speed [rpm]	Torque [Nm]	Tool wear [min]	Machine failure	TWF	HDF	PWF	OSF	RNF
1	298.1	308.6	1551	42.8	0	0	0	0	0	0	0
2	298.2	308.7	1408	46.3	3	0	0	0	0	0	0
3	298.1	308.5	1498	49.4	5	0	0	0	0	0	0
4	298.2	308.6	1433	39.5	7	0	0	0	0	0	0
5	298.2	308.7	1408	40	9	0	0	0	0	0	0
6	298.1	308.6	1425	41.9	11	0	0	0	0	0	0
7	298.1	308.6	1558	42.4	14	0	0	0	0	0	0
8	298.5	309	1741	28	21	0	0	0	0	0	0

2.3 Model Design

The model uses Unified Modelling Language (UML) to visually represent the research methodology, making it easier to communicate and standardise across projects. According to Gadhi et al. (2023), UML diagrams help identify potential bottlenecks and improvements before implementation.

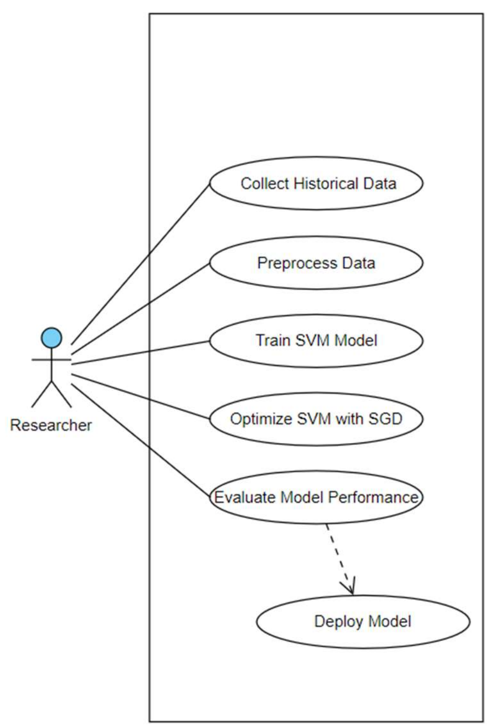

Figure 2: Use case diagram of the model

Figure 2, Use Case Diagram, outlines interactions between the actor and system components, covering processes like designing, training, and optimising the SVM algorithm. The Activity Diagram, Figure 3 illustrates the research workflow from data collection to model deployment. The Class Diagram, Figure 4, shows key components like the researcher, dataset, SVM model, optimiser, and evaluation

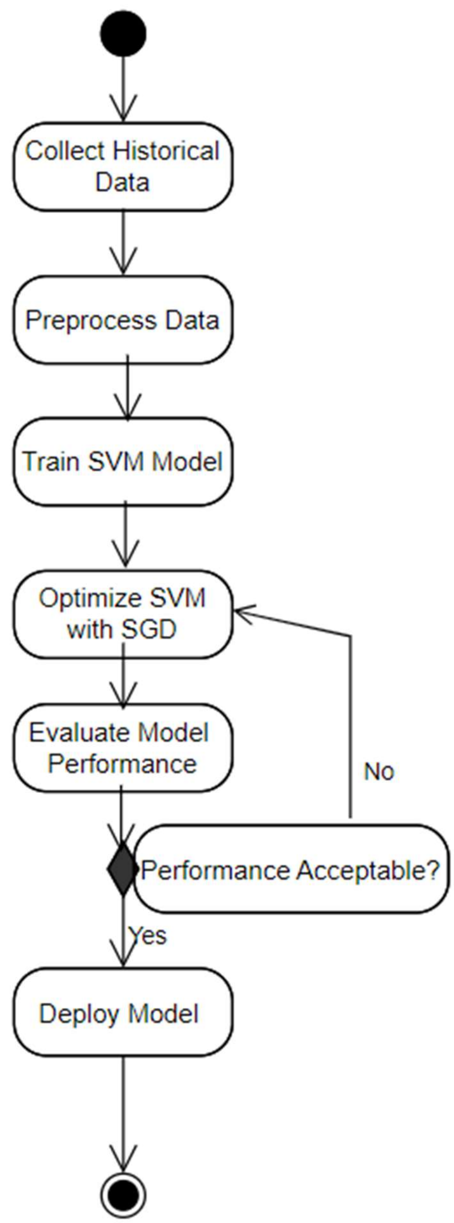

Figure 3: Activity diagram of the model

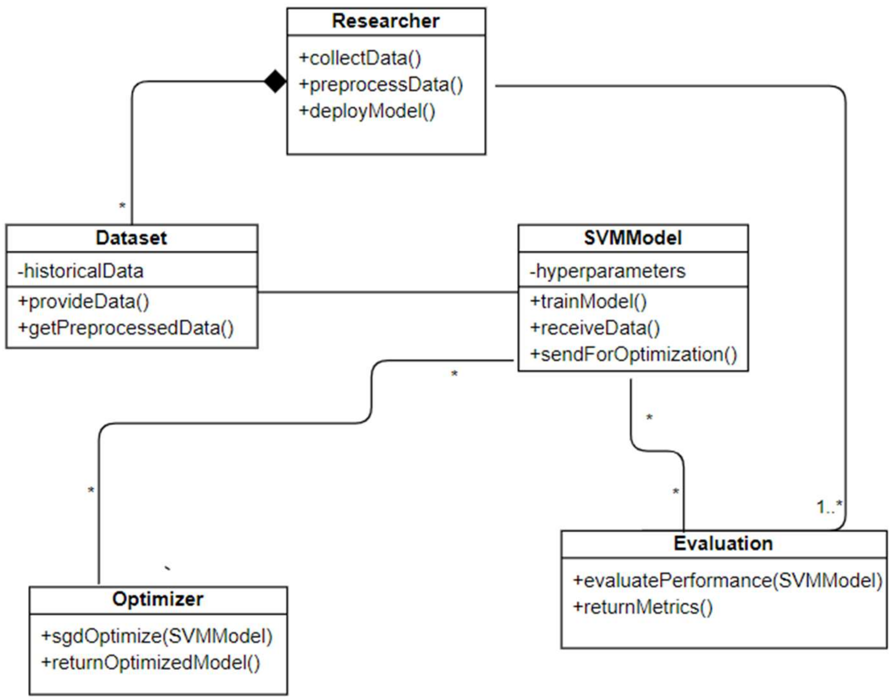

Figure 4: Class Diagram of the Model

3. MODEL FORMULATION: SVM MODEL FOR EQUIPMENT FAILURE PREDICTION

The SVM is applied to equipment failure classification, aiming to predict failures using the labelled dataset, DDD, defined as follows:

$D = \{(x_i, y_i) \mid x_i \in \mathbb{R}^p, \ , y_i \in \{0, 1\}\} i=1, 2..., n$ ....(1)

Where:

- x_i represents each feature vector (e.g., "Air Temperature [K]", "Process Temperature [K]", "Rotational Speed [rpm]", "Torque [Nm]", "Tool Wear [min]").
- y_i is the class label for "Machine Failure", where 1 indicates failure, and 0 indicates no failure.

3.1 Hyperplane Function

The hyperplane function g(x) that classifies equipment failure is given by:
g(x) = $\acute{\omega}^T x$ + b ....(2)

Where:

- $\acute{\omega}$ is the weight vector corresponding to the features x.
- b is a scalar bias term
- T indicates the transposition of the vector

For the linearly separable case, the hyperplane satisfies:

- g(x) > 0 for failure cases (y_i = 1)
- g(x) < 0 for failure cases (y_i = 0)

To address non-linear decision boundaries, the input space is mapped to a higher-dimensional space using a kernel function K(x_i, x_j) to facilitate linear separability in the transformed space.

3.2 Hyperparameter Tuning Using Grid Search

To optimise the model, the **grid search** method was used to tune SVM hyperparameters such as the regularisation parameter C and kernel function (e.g., radial basis function).

The tuning objective is to minimise the classification error by maximising the margin between the classes. The hyperparameter tuning objective function is:

$$\min_{\alpha} \left(\frac{1}{2}\sum_{i=1}^{n}\sum_{j=1}^{n} \propto_i \propto_j y_i y_j K(xi, xj) - \sum_{i=1}^{n} \propto_i\right) \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (3)$$

Where:

- α represents the Lagrange multipliers
- y_i represents the class labels (0 or 1 for non-failure or failure)
- K(x_i, x_j) is the kernel function applied to input features.

3.3 Optimization with Stochastic Gradient Descent (SGD)

After tuning, Stochastic Gradient Descent (SGD) further optimises the model, iteratively updating parameters to speed up convergence. The SGD update rule for the SVM parameters θ=(ω,b) is:

$$\theta = \theta - \eta . \nabla_\theta J(\theta) \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (4)$$

Where:

- η is the learning rate.
- $\nabla_\theta J(\theta)$ is the gradient of the hinge loss function with respect to the model parameters θ.

This optimisation process minimises the classification error and enhances the SVM's ability to generalise to new failure data.

Algorithm 1: Algorithm of SVM-SGD

```
function SGD_SVM (X_train, y_train, C, alpha, num_iterations):
    # Initialize weights and bias
    theta = initialise zeros of size m (features)
    bias = 0
    n, m = dimensions of X_train
        # SGD optimisation loop
    for iteration in range(num_iterations):
        shuffle (X_train, y_train)  # Shuffle data for stochasticity
            for i in range(n):
          # Calculate margin for SVM
          margin = y_train[i] * (dot_product(theta, X_train[i]) + bias)
                if margin < 1:  # Hinge loss condition
            # Update weights and bias using gradient descent
            theta -= alpha * ( -y_train[i] * X_train[i] + 2 * (1 / (C * n)) * θ )
            bias  -= alpha * (-y_train[i])  # Bias update when margin is violated
          else:
            # Update only regularisation term
            theta -= alpha * (2 * (1 / (C * n)) * theta)
    return theta, bias  # Optimized weights and bias
function predict(X_test, theta, bias):
    return [sign(dot_product(theta, x) + bias) for x in X_test]  # Predict labels
```

3.4 Key Points of SVM-SGD:

Margin check: If margin < 1, the SVM experiences a violation (hinge loss), and the weights θ and bias b are updated using gradient descent.

Regularisation: The term 2 * (1 / (C * n)) * θ ensures regularisation, controlling overfitting by penalising large weights.

Stochastic updates: The SGD optimises the SVM by updating the weights incrementally for each training example, leading to fast convergence.

This algorithm captures where the SVM is optimised whenever the margin violates the hinge loss condition (i.e. when margin < 1).

3.5 Model Performance Evaluation

The following metrics were used in assessing the model performance: precision, recall, F1 score, and accuracy. These metrics are calculated thus.

$$Precison = \frac{TP}{TP+FP} \quad (5)$$

$$Recall = \frac{TP}{TP+F} \quad (6)$$

$$F1 - Score = 2 * \frac{Precision * Recall}{Precision + Recall} \quad (7)$$

$$Accuracy = \frac{TP + TN}{TP + FN + FP + FN} \quad (8)$$

Where TP = True positive, FP = False positive, TN = True negative, FN = False negative.

4. RESULT AND DISCUSSION

The dataset contains 10,000 instances, with 9,661 labelled as 'no failure' and 339 as 'failure,' reflecting a significant imbalance. This imbalance is a common challenge in predictive maintenance, as failure events are rare compared to normal operations Dangut et al. (2023). The SVM model achieved a recall of 0.57, indicating that 57% of failures were correctly identified. With an accuracy of 0.78, the model primarily predicts non-failures due to the dataset imbalance. An AUC-ROC of 0.74 indicates moderate model performance, particularly to reduce false positives and enhance prediction reliability.

The optimised SVM model, enhanced with SMOTE and random under-sampling to handle class imbalance, demonstrated strong performance. A recall of 0.80 indicates that 80% of actual failures were correctly identified, which is important for proactive maintenance. While the precision was 0.37, highlighting a trade-off with recall, this is acceptable in maintenance, where capturing failures is a priority Paroha et al. (2024). With 95% accuracy and an AUC-ROC of 0.96, the model effectively distinguishes between failure and non-failure events, supporting robust predictive maintenance.

Table 2: Comparison of the SVM model and Optimised SVM model performances

Metric	SVM Model	Optimised SVM Model
Recall	0.57	0.80
Precision	0.08	0.37
F1-Score	0.14	0.50
Accuracy	0.78	0.95
AUC-ROC	0.74	0.96

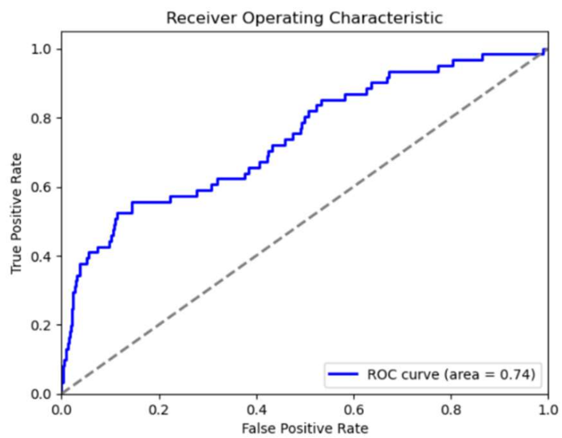

Figure 5: The AUC-ROC Graph for the SVM Model

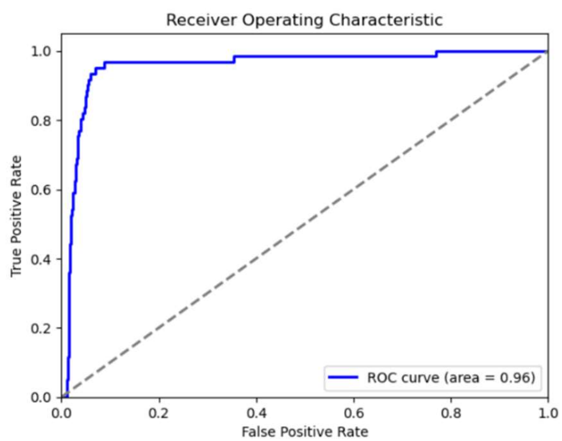

Figure 6: The AUC-ROC Graph for the Optimised SVM Model

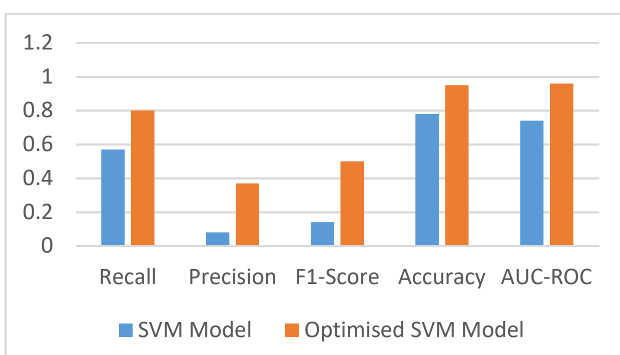

Figure 7: Bar-chart comparison of the SVM and Optimised SVM models performances

The significant class imbalance between failure and non-failure instances affected the SVM model's precision, leading to false positives. While the model achieved moderate recall, this came at the cost of lower precision, meaning many non-failures were misclassified as failures. The accuracy (0.78) and AUC-ROC (0.74) indicated reasonable performance, but further refinement is needed to balance recall and precision. To address this, SMOTE and random undersampling were used to tackle the imbalance, and grid search with Stochastic Gradient Descent (SGD) optimised the SVM's hyperparameters. These techniques improved the model's recall and AUC-ROC scores, enhancing its ability to identify failures effectively. Despite the trade-off in precision, the model aligns well with predictive maintenance priorities, where detecting failures is crucial.

5. CONCLUSION

This study developed an optimised predictive model for equipment failure and maintenance using Support Vector Machines (SVMs). The optimised SVM model, enhanced through various techniques, strongly predicted equipment failures. With a recall of 0.80 and an AUC of 0.96, the model effectively distinguished between failure and non-failure cases, underscoring its potential for practical applications in predictive maintenance. The findings of this study indicate that the optimised SVM model can significantly improve predictive maintenance practices by reducing downtime and maintenance costs. The high recall rate ensures that most equipment failures are detected early, allowing for proactive maintenance scheduling. However, the relatively low precision could suggest more false positives, which could lead to unnecessary maintenance activities. Despite this limitation, the model's high accuracy highlights its utility in enhancing maintenance strategies.

REFERENCES

Goswami, P., and Rai, R. N. (2023). A Systematic Review on Failure Modes and Proposed Methodology to Artificially Seed Faults for Promoting PHM Studies in Laboratory Environment for an Indus Rial Gearbox. *Engineering Failure Analysis*, 107076.

Sezer, E., Romero, D., Guedea, F., Macchi, M., and Emmanouilidis, C. (2018). An industry 4.0-enabled low-cost predictive maintenance approach for SMEs. In *2018 IEEE International Conference on Engineering, Technology and Innovation (ICE/IT C)* (pp. 1-8). IEEE.

Ren Y. (2021). Optimising predictive maintenance with machine learning for reliability improvement. *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part B: Mechanical Engineering*, *7*(3), 030801.

Wang, J., and Gao, R. X. (2022). Innovative smart scheduling and predictive maintenance techniques. In *Design and Operation of Production Networks for Mass Personalization in the Era of Cloud Technology* pp. 181-207). Elsevier.

Zhong, R. Y., Xu, X., Klotz, E., and Newman, S. T. (2017). Intelligent manufacturing in the context of industry 4.0: a review. *Engineering*, *3*(5), 616–630.

Leukel, J., González, J., and Riekert, M. (2021). Adoption of machine learning technology for failure prediction in industrial maintenance: A systematic review. *Journal of Manufacturing Systems*, *61*, 87-96.

Lee, J., Ni, J., Singh, J., Jiang, B., Azamfar, M. and Feng, J., (2020). Intelligent maintenance systems and predictive manufacturing. *Journal of Manufacturing Science and Engineering*, *142*(11), p.110805.

Kotsiopoulos, T., Sarigiannidis, P., Ioannidis, D., and Tzovaras, D. (2021). Machine learning and deep learning in intelligent manufacturing: The smart grid paradigm. *Computer Science Review*, *40*, 100341.

Campos, J. R., Vieira, M., and Costa, E. (2018). An exploratory study of machine learning techniques for supporting failure prediction. In *2018 14th European Dependable Computing Conference (EDC)* (pp. 9-16). IEEE.

Mohammed, B., Awan, I., Ugail, H., and Youn s, M. (2019). Failure prediction using machine learning in a virtualised HPC system application. *Cluster Computing*, *pp. 22*, 471–485.

Celikmih, K., Inan, O., and Uguz, H. (2020). Failure prediction of aircraft equipment using machine learning with a hybrid data preparation method. *Scientific Programming*, *2020*, 1-10.

Jimenez, V. J., Bouhmala, N., and Gausdal, A. H. (2020). Developing a predictive maintenance model for ves el machinery. *Journal of Ocean Engineering and Science*, *5*(4), 358-386.

Luís, F., André, P., Vítor, S., Filipe, R., and Paulo, C. (2021). Prediction of Maintenance Equipment Failures Using Automated Machine Learning. *Intelligent Data Engineering and Automated Learning – IDEAL 2021*. Springer International Publishing

Shaheen, A., Hammad, M., Elmedany, W., Ksantini, R., and Sharif, S. (2023). Machine failure prediction using joint reserve intelligence with feature selection technique. *International Journal of Computers and Applications*, pp. 45, 638 - 646.

Gadhi, A., Gondu, R. M., Bandaru, C. M., Reddy, K. C., and Abiona, O. (2023). Applying UML and Machine Learning to Enhance System Analysis and Design. *International Journal of Communications, Network and System Sciences*, *16*(5), 67-76.

Dangut, M. D., Jennions, I. K., King, S., and Skaf, Z. (2023). A rare failure detection model for aircraft predictive maintenance using a deep hybrid learning approach. *Neural Computing and Applications*, *35*(4), 2991-3009.

Paroha, A. D. (2024, April). Advancing Predictive Maintenance in the Oil and Gas Industry: A Generative AI Approach with GANs and LLMs for Sustainable Development. In *International Conference on Sustainable Development through Machine Learning, AI and IoT* (pp. 3-15). Cham: Springer Nature Switzerland.