ISSN No:-2456-2165

Maximising Operational Uptime: A Strategic Approach to Mitigate Unplanned Machine Downtime and Boost Productivity using Machine Learning Techniques

Authors: Asit Das[1], Karanam Srihari [2], Susmitha Pasupuleti[3], Indira Kumar[4], Bharani Kumar Depuru[5]
[1,2]Research Associate, Innodatatics Inc., Hyderabad, India
[3]Mentor, [4]Team Leader, Research and Development, Innodatatics Inc., Hyderabad, India
[5]Director, Innodatatics Inc., Hyderabad, India

*Corresponding Author: Bharani Kumar Depuru
ORC ID: 0009-0003-4338-8914

Abstract:- In the sphere of industrial processes, the constant operation of machinery is paramount, and any downtime resonates with substantial losses in productivity, efficiency, and profitability. The industry confronts the intricate challenge of minimizing machine downtime attributed to breakdowns, unscheduled maintenance, operator errors, and environmental factors. This predicament creates a cascade of adverse effects, diminished efficiency, missed production targets, heightened maintenance costs, and reduced profitability.

This research paper charts a strategic roadmap designed to address the challenge of minimizing unplanned machine downtime. With a focus on key objectives, including maximizing machine productivity, minimizing downtime through root cause identification and preventive measures, reducing maintenance costs, and ultimately enhancing overall efficiency for improved competitiveness and profitability. However, a set of constraints introduces complexity to the implementation of these objectives. Budgetary constraints, time limitations, resource scarcity, regulatory requirements, and operational constraints intricately weave a tapestry that demands thoughtful navigation.

The proposed strategy encompasses a multifaceted approach integrating preventive measures, root cause analysis, and efficiency optimization. The paper navigates through these strategies, taking into account the identified constraints, offering a holistic framework to enhance machine reliability and performance. Through a judicious balance of technological innovation, preventive maintenance, and operational optimization, the industry aspires to revolutionize manufacturing processes, mitigate downtime challenges, and emerge as a more competitive and profitable entity in the industrial domain.

Moreover, in the realm of machine downtime classification, this study employs diverse models such as Logistic Regression, Naive Bayes, K-Nearest Neighbors (KNN), Decision Tree, AdaBoost Classifier, Gradient Boosting, Random Forest, Extra Tree Classifier, and HistGradient Boosting. The evaluation criteria include accuracies, recall, precision and F1 scores, offering a comprehensive assessment of each model's effectiveness in predicting and preventing machine downtime. Notably, Random Forest outperforms other models, adding a significant layer of insight for industries seeking efficient measures in machine downtime management.

Keywords:- *Machine Downtime Prediction, Python, PowerBI, Machine learning models, Streamlit, Predictive Maintenance, Operational Uptime Optimization, Industrial Equipment Efficiency, Production Loss Prevention, Manufacturing Productivity Enhancement.*

I. INTRODUCTION

Equipment used in the manufacturing field is frequently used without a planned maintenance strategy due to certain unanticipated failures; such a practice frequently causes unexpected downtime [1]. This study aims to enhance job performance by implementing strategic measures to minimise unscheduled machine time and boost overall efficiency. The research uses advanced technologies such as predictive maintenance, predictive error modelling, and machine learning algorithms to perform an in-depth analysis of machine history data. Specifically, the study investigates the effectiveness of models such as Random forests and ensembles in predicting and preventing outliers. This study aims to understand how the impact of unplanned technology on production can be reduced by evaluating the effectiveness of these advanced technologies.

In manufacturing, an unplanned machine causes serious problems by affecting production plans, increasing maintenance costs and increasing total product losses [2]. This study recognizes the urgent need for effective strategies to solve this problem and discover new ways to predict and prevent machine failure. The importance of using data-driven insights and advanced technology to develop a strong foundation for minimising unplanned downtime and thus increasing operational uptime [3].

As businesses strive for automation and efficiency, unplanned machine downtime is becoming more common. Research parallels advancements in business processes where the integration of smart tools and predictive analytics is vital for better competitive management. By using the reduction method, companies can not only increase efficiency but also increase the competitiveness of companies by achieving long-term savings [4].

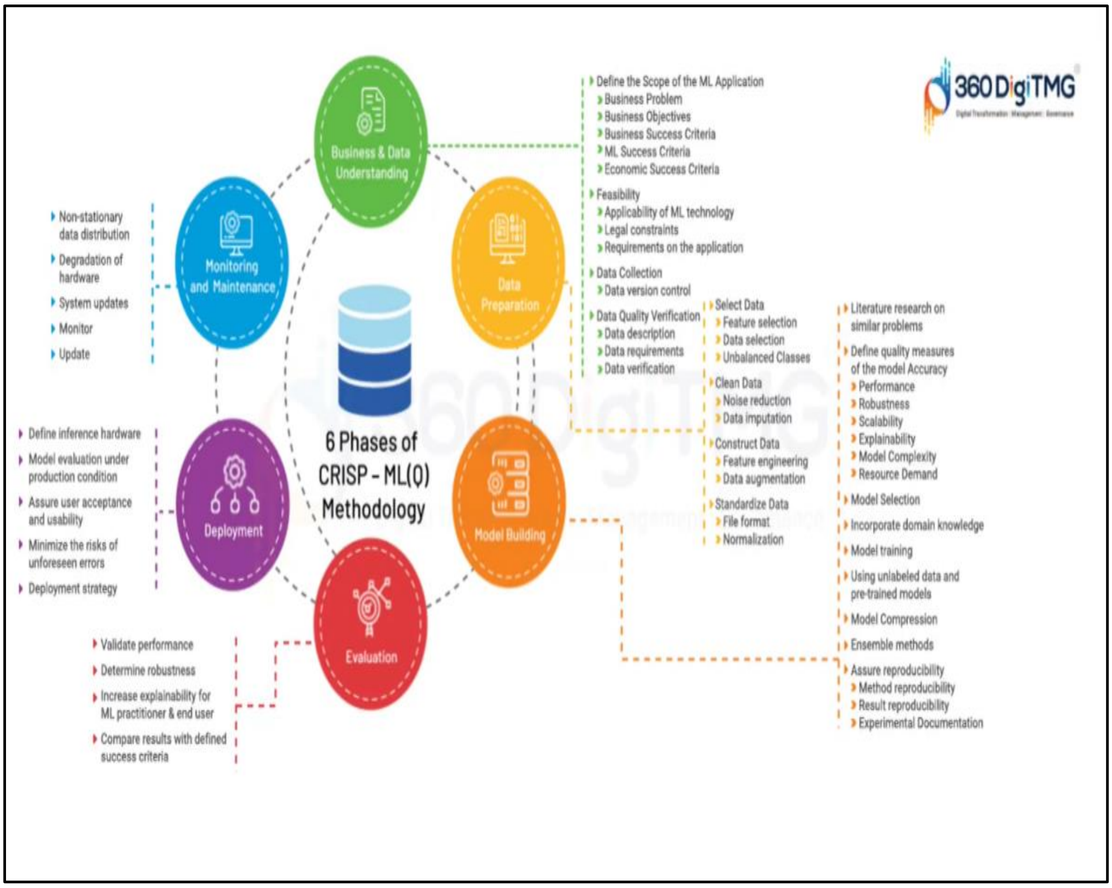

Fig.1: The CRISP-ML(Q) Methodological Framework offers a visual roadmap of its integral components and sequential steps. (Source:- Mind Map - 360DigiTMG)

In the initial stage of CRISP-ML(Q) [Fig.1], the primary objective is to comprehend industry dynamics. The framework guides machine learning development, prioritising quality assurance [5]. This study showcases the impact [Fig.1] and innovative functionality [Fig.2] of the CRISP-ML(Q) approach, detailing a step-by-step process to address challenges in unplanned machine downtime.

The initial step, "Business Understanding" [Fig.1][6], involved comprehending the goals and requirements associated with managing machine downtime. The aim was to recognize the impact of unplanned downtime on efficiency, production targets, maintenance costs, and overall profitability was pivotal.

Entering the "Data Understanding" phase [Fig.1], we meticulously gathered and analysed relevant datasets for insights into factors influencing machine downtime. This involved examining historical data on breakdowns, maintenance, operator performance, and environmental conditions [7]. Our methodology began with a thorough exploration of machine data from January 2022 to January 2023, covering attributes such as Date, Machine_ID, Load cells, Hydraulic Pressure, Coolant Pressure, Air System Pressure, Coolant Temperature, Hydraulic Oil Temperature, Proximity sensors, Spindle Vibration, Tool Vibration, Spindle Speed, Voltage, Torque, Cutting Force, and Downtime.

In the "Data Preparation" phase [Fig.1], our attention was dedicated to refining the gathered data. This involved meticulous pre-processing to guarantee its reliability and appropriateness for modelling purposes. Tasks encompassed addressing missing values, cleansing the data, and executing essential variable transformations [8]. Additionally, we employed feature engineering techniques to extract pertinent predictors, thereby augmenting the predictive capabilities of the models.

In "Data Mining" [Fig.1], the process involves gathering, cleaning, processing, analysing, and distilling practical insights from the data. Advancing to the "Model Building" [Fig.1] stage, we employ sophisticated machine learning [9] approaches, particularly leveraging models like Random Forest, Decision tree, KNN, Naive Bayes and Ensemble methods [10]. These models excel in discerning intricate connections and trends within the data, facilitating the development of reliable and precise models geared towards preventing unplanned downtime.

ISSN No:-2456-2165

In the "Model Deployment" [Fig.1] phase, the generated models were integrated into the machine management system. This integration facilitates the optimization of machine downtime. These findings provide actionable recommendations to stakeholders such as machine operators, maintenance teams and decision makers to improve efficiency and allocate resources optimally in response to unplanned situations.

The well-known CRISP-ML(Q) method in data mining forms the basis of this study. This approach has been adopted as the de facto standard for knowledge and information research activities, ensuring efficiency and focusing on the effectiveness of research on various models in predicting and preventing machine failures.

- **Architecture Diagram:**

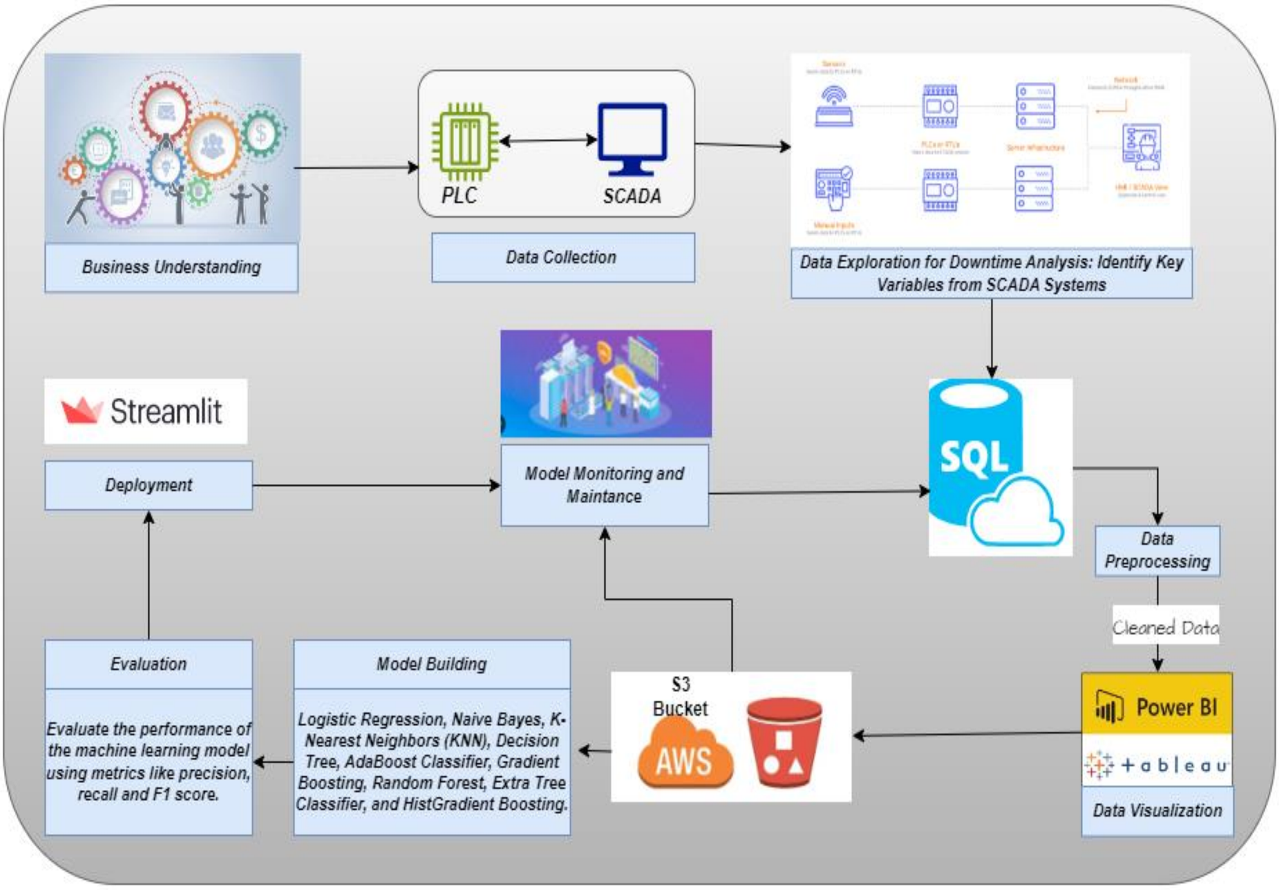

Fig. 2: Comprehensive project flow depicted through an architectural diagram.
(Source: ML Workflow - 360DigiTMG)

II. METHODOLOGY AND TECHNIQUES

A. Data collection

The data used in this study has been provided by an esteemed client, a leading manufacturing company based in Germany. The primary objective of this study is to improve operational efficiency and minimise unplanned machine downtime for an automatic stirrup bender. The data contains various features which serve as an important resource for understanding the complexity of the manufacturing process.

- *Data dimension:*

Presenting key information about our Machine Downtime Analysis Dataset.

Data Size	461 KB
Number of Records	2,700
Features	16
Data Type	Numeric features: 14 Categorical features: 2
Format	Comma-separated values (CSV)

Provided herein the data that is required for our study is detailed below focusing on specific sensor details and SCADA (Supervisory Control and Data Acquisition) software integration [11].

- *Determine the machine type:*

This study focuses on a specific automatic stirrup bending machine and requires a clear understanding of its functions and material procedure. In the automation industry, the integration of various sensors with SCADA (Supervisory Control and Data Acquisition) and PLC (Programmable Logic Controller) [11] is essential for monitoring and controlling time. These stand-alone devices, including temperature and pressure data loggers, connect to machine sensors [12] to collect data over time. This study proposes a temporal system that interconnects temperature, pressure, proximity and angle to ensure accurate data collection and efficient operation.

- *Sensor selection and its connectivity:*

Select temperature sensors for capturing thermal readings, pressure sensors for hydraulic details, proximity sensors for object detection and rotation angle measurements. Establish a robust connection between the PLC and SCADA system to implement a dependable communication protocol to facilitate smooth interaction. Ensure the effectiveness of interaction between the PLC and SCADA for optimal system integration [11, 13]. Communicate between two ends to create a stable and secure connection. Monitor real-time data on the SCADA interface and see how the PLC responds to inputs from temperature, pressure, proximity and angle sensors.

- *Software Systems:*

The integration of SCADA software creates a core hub for industrial automation, coordinating multiple sensors that are essential for data capture and control. These sensors, encompassing Load Cells, Hydraulic Pressure, Coolant Pressure, Air System Pressure, Coolant Temperature, Hydraulic Oil Temperature, Proximity Sensors, Vibration Sensors (Spindle and Tool), Speed Sensors (Spindle Speed), Voltage Sensor, Torque Sensor, Cutting Force Sensor, and Downtime indicator, collectively contribute to a sophisticated network.

SCADA software serves as the backbone, coordinating the communication and analysis of data from these sensors [14]. For instance, Load Cells offer insights into machine load, Hydraulic Pressure and Coolant Pressure sensors monitor fluid dynamics, and Proximity Sensors track spatial relationships. Vibration Sensors assess spindle and tool vibrations, the Speed Sensor measures spindle rotations, and the Voltage Sensor monitors electrical inputs. This seamless integration ensures precise and real-time data analysis, optimising machine downtime, fostering proactive maintenance, and enhancing overall operational efficiency in industrial settings.

B. Data Preprocessing:

This section provides information to understand business issues before implementing the data modelling. It includes phases such as data cleaning, data transformation and data selection [8]. Data preparation tasks can be done multiple times not in any particular order in this stage. Crucial matters such as identifying pertinent data, organising data and eliminating unacceptable data are addressed during this phase. In ensuring optimal dataset quality, we meticulously address anomalies [15].

Duplicate entries were systematically identified and removed, averting redundancy. For missing data points, robust imputation techniques were adeptly employed, ensuring precise estimations. Alignment of data types and standardisation of numerical features were seamlessly executed, preventing biases. Outliers underwent strategic detection and management, ensuring robust data integrity. Feature engineering was applied judiciously, introducing insightful elements based on existing patterns in machine performance and production logs. Categorical data underwent seamless label encoding, facilitating integration into machine learning models.

Quality assurance checks were implemented, validating the accuracy and reliability of the pre-processed data [16]. The dataset was efficiently split for training and testing, a pivotal step for evaluating model performance on new data. This comprehensive yet succinct data preprocessing ensures the dataset is finely tuned, setting the stage for effective machine learning model implementation and ultimately leading to accurate prevention of unplanned machine downtime.

C. Data Pipeline

Maintaining the data pipeline's adaptability to the ever-changing data landscapes is made possible by regular updates and ongoing monitoring. Retaining consistency in raw data processing through a strong data pipeline is essential to improve quality, flexibility and repeatability [17]. The proposed data pipeline aims to optimise the handling of unprocessed data, extract valuable features and apply ML models. It is specifically tailored to tackle issues related to machine downtime. As demonstrated in [Fig.3] the suggested data pipeline functions as a reliable framework and can be adjusted to suit different situations.

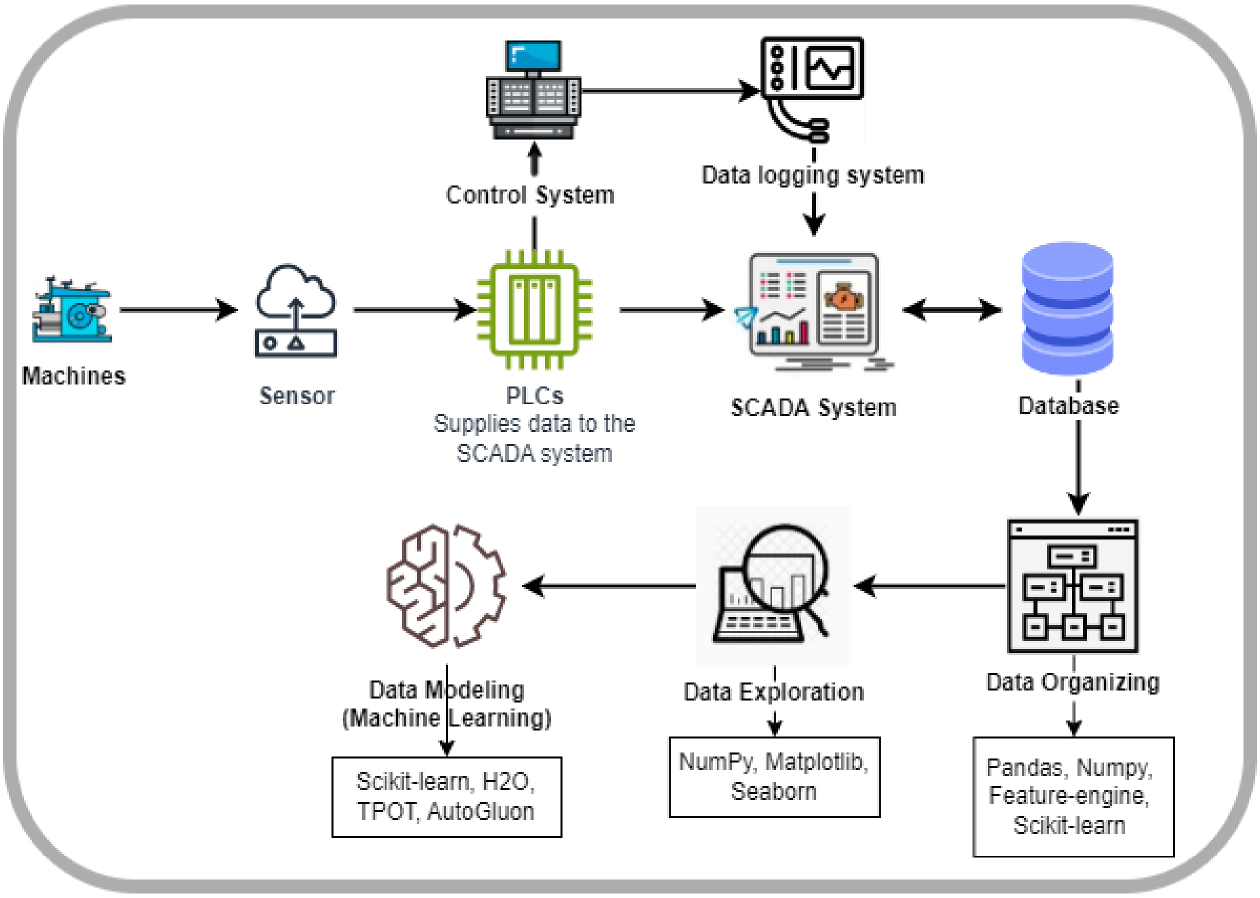

Fig.3: Efficient Data Pipeline Design for Machine Downtime Analysis in Industrial Automation

Raw data is gathered from various sensors and centralised during the first data acquisition step in preparation for additional processing. Python is the only language used for the data acquisition process, so information about machine downtime is integrated seamlessly. Next comes a data wrangling (data organising) step that uses the pandas, numpy library in Python to convert the data into a standard format [6]. This includes procedures for cleaning, reduction and integration. In order to properly prepare the dataset for analysis, this step is essential. Additionally, the initial descriptive analysis and visualisation carried out during the data exploration phase, powered by Python's matplotlib and seaborn, reveal patterns and insights pertaining to machine downtime [18]. The next stage of data modelling is to create prediction models by using scikit-learn in Python to predict and handle possible machine failure scenarios.

D. *Exploratory Data Analysis (EDA)*

In the realm of exploratory data analysis (EDA), a fundamental step in the research process, our focus delves into comprehending the intricate characteristics of the dataset [19]. While there are no rigid rules for EDA, common approaches encompass summary statistics, correlation analysis, visualisation, and aggregation techniques [6]. Specifically, our investigation revolves around understanding the machine environment, unravelling machine dynamics and performance, scrutinising statistical characteristics and variability, and discerning distribution patterns.

➤ *Understanding the Machine Environment:*

Data interpretation starts with summary statistics, also known as univariate analysis providing a fundamental approach to grasp the characteristics of individual variables(features/ attributes) in the dataset [20]. Our initial exploration reveals that, on average, machining operations involve a relatively low applied force, signifying a trend towards moderate force applications. Concurrently, coolant pressure maintains a consistently low profile, indicating a persistent requirement for temperature regulation. Air system pressure and coolant temperature underscore moderate operational demands in these facets.

➤ *Machine Dynamics and Performance*

Delving deeper, we examine the dynamics and performance of the machine. Vibration levels in the spindle and tool provide critical insights into the stability of the machining process. The spindle's relatively high rotation speed influences the efficiency and pace of machining operations [Fig.4(a)]. High electrical voltage requirements underscore the substantial power demand for the machine. Additionally, torque values signify the rotational force applied to the spindle, revealing the machining power. Cutting force levels indicate a moderate force typically exerted during machining [Fig.4(b)].

➤ *Statistical Characteristics and Variability*

The analysis further extends to scrutinising statistical characteristics and delving into the variability present within the dataset. This includes a meticulous examination of fluctuations, patterns, and trends [Fig.5], providing a comprehensive view of the data's inherent variability and shedding light on potential factors influencing its diverse patterns [20]. Hydraulic pressure displays considerable variability, implying fluctuations in power demands. Temperature readings exhibit varying degrees of consistency, impacting the thermal conditions of the machine. Vibration levels demonstrate varying degrees of consistency, directly affecting the stability of the machining process. Spindle speed, electrical voltage, torque, and cutting force each showcase distinct levels of variability.

➢ *Distribution Characteristics:*

Metrics like skewness which represents the degree of symmetry, and kurtosis which represents the properties of the tails, are crucial in the field of exploratory data analysis (EDA) [6]. Although kurtosis provides information about the dataset's tail behaviour, skewness explores the dataset's asymmetry.In exploring distribution patterns, we find that force applied, hydraulic pressure, and air system pressure exhibit right-skewed distributions with high peaks and heavy tails. This suggests the prevalence of lower values with occasional higher values, highlighting susceptibility to extreme values, particularly in the case of cutting force. This comprehensive research-oriented analysis lays the groundwork for deeper investigations into the intricacies of machine dynamics and guides further research directions for optimization and enhancement in industrial operations.

➢ *Statistical Analysis*

Furthermore, we construct correlation matrices to conduct multivariate Exploratory Data Analysis (EDA) [6]. This involves assessing the correlation coefficients between different variables providing valuable insights into the relationships within the dataset.

- Certain sensing elements, particularly those gauging physical loads and proximity, demonstrate significant correlations with vibrational aspects (related to the machine's spindle and tool) as well as cutting force. These associations imply similarities in measurements or shared underlying factors influencing these readings.
- Within the hydraulic components, observations indicate noteworthy correlations between pressure measurements (both hydraulic and coolant) and temperature readings (coolant and oil). This interdependence reflects a systemic relationship among these variables.
- In the realm of dynamic interactions, aspects related to motion (spindle speed) exhibit discernible connections with measures of vibration and force, specifically in the context of the tool and cutting forces. This indicates a tendency for heightened force and vibration under increased speed.
- In another domain, a subtle negative correlation emerges between a rotational force (torque) and the temperature of a cooling component. This implies that elevated torque might contribute to a marginal cooling effect.
- Conspicuously standing apart, a certain factor demonstrates no significant associations with other variables in the analysis. This suggests its relative independence and minimal impact on the overall process.

The most influential features for modelling, based on their significance and impact, can be discerned by considering their correlation strengths, variability, and potential predictive power. Features that exhibit strong correlations with the target variable and display substantial variability are often crucial for modelling [21]. Additionally, features with high predictive power, as indicated by their impact on the output or target, are influential. The features related to force application, hydraulic dynamics, and sensor measurements appear to be influential.

The identified influential features, especially those pertaining to force application, hydraulic dynamics and sensor measurements, provide a strong basis for further model development, building on the extensive exploratory data analysis(EDA). Important insights into the complexities of the industrial processes under investigation are provided by the observed correlations and dependencies among these features. With a clear understanding of the dataset’s characteristics and influential variables, the next steps involve formulating and implementing a targeted model that can harness these insights for predictive accuracy and operational optimization.

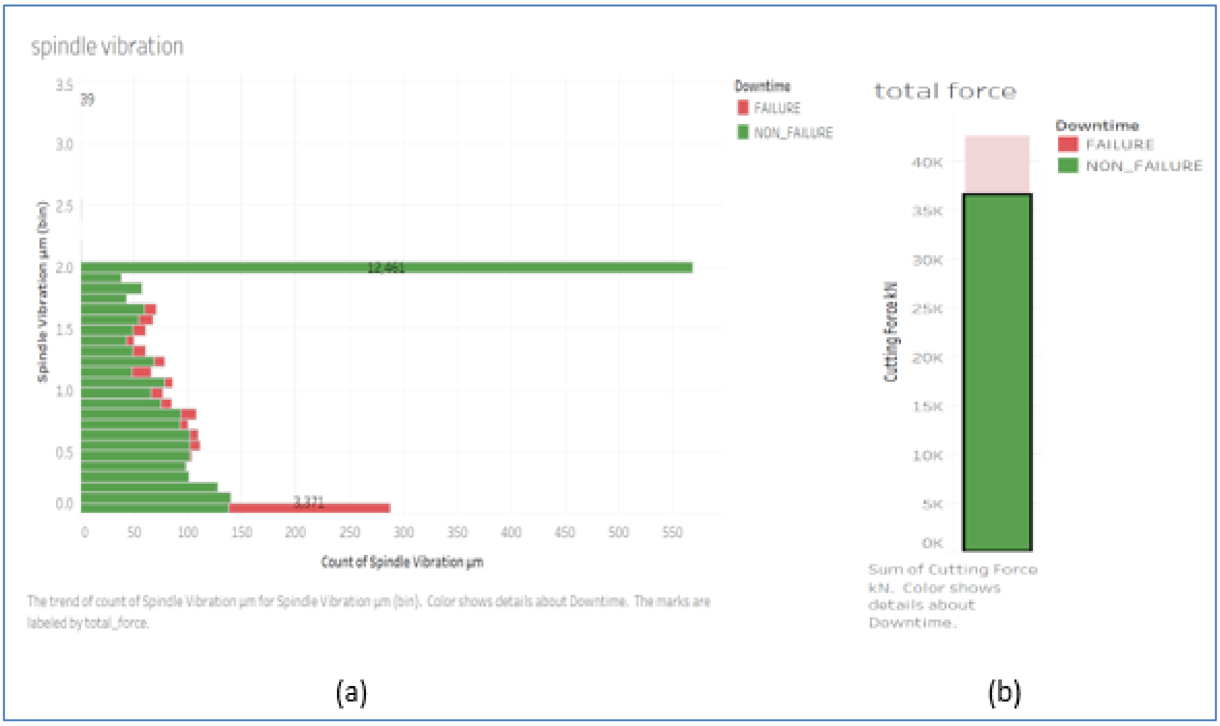

Fig. 4: Tool Performance Analysis
(a) Vibration measurements of a spindle over time
(b) Tool Efficiency with Cutting Force Data

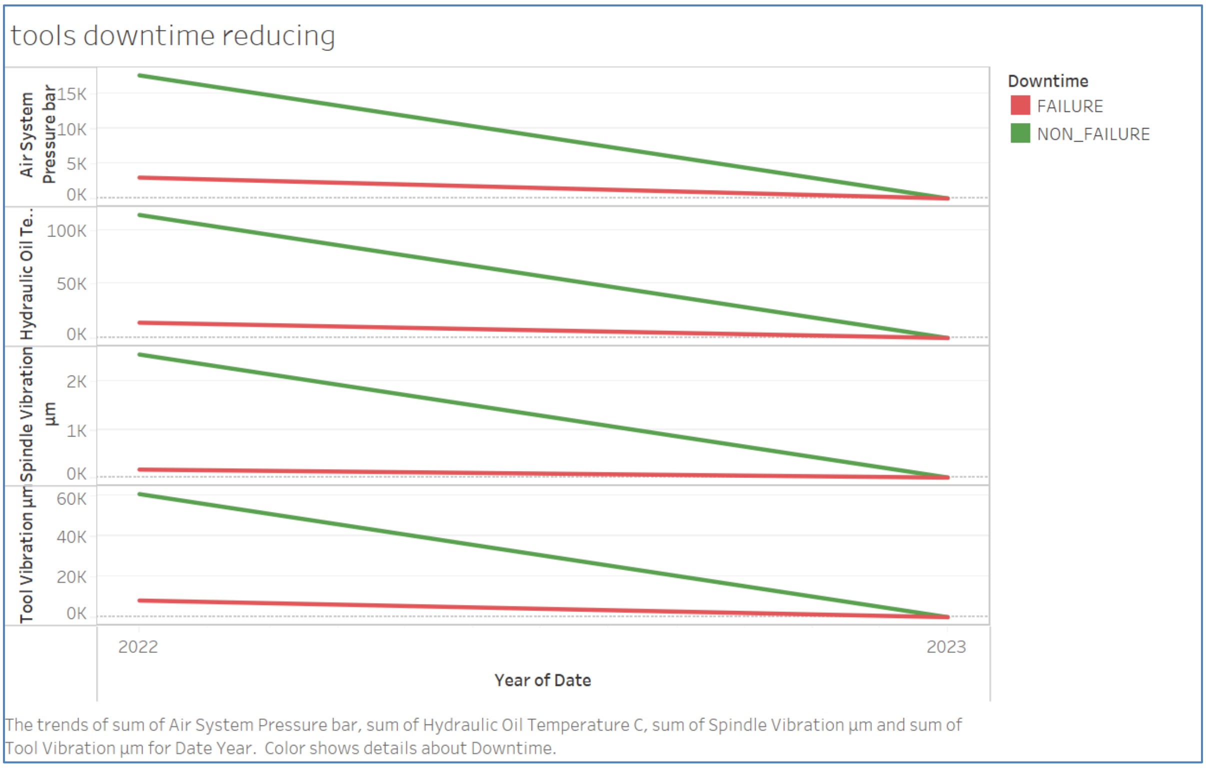

Fig. 5: Machine tools performance trends over time

- *Data Splitting*

Algorithms learn from training data in order to produce predictions or well-informed decisions during the testing and training stages. In order to properly evaluate the performance of an algorithm, datasets are divided into two subsets: a training set and a testing or validation set [15,22]. With the remaining 20% going toward testing and validation, the training set—which makes up 80% of the original dataset—provides a strong basis for algorithm training.

```
X_train, X_test, y_train, y_test = train_test_split(X, Y, test_size=0.2)
```

This code showcases the use of the `train_test_split` function, a vital component of the evaluation process. It divides the dataset into `X_train` and `X_test` for feature matrices, as well as `y_train` and `y_test` for respective target variables. This strategic partitioning ensures a dedicated portion is set aside for unbiased testing and validation.

E. Model Approach

Our approach to optimising machine downtime for enhanced productivity involves leveraging advanced classification models. Specifically, we use cutting-edge classification models to optimise machine downtime for increased productivity. Productivity suffers directly from machine downtime or the time a machine is not in use. In order to tackle this challenge, we have employed several classification models, such as Logistic Regression, Naive Bayes, K-Nearest Neighbors (KNN), Decision Tree, AdaBoost Classifier, Gradient Boosting, Random Forest, Extra Tree Classifier, and HistGradient Boosting [23].

These models, which analyse past data, are essential for anticipating possible machine faults. Their capacity to classify machines into operational and potential failure stages gives us the opportunity to plan maintenance proactively. We are able to schedule maintenance preemptively because of their capacity to categorise machines into functioning and probable failure statuses. Choosing categorization models calls for strategic consideration. The ability of these models to identify patterns and occurrences prior to downtime allows for early intervention. By foreseeing issues before they worsen, we can better manage resources, reduce unplanned downtime, and enable a more efficient production environment, all of which increase total productivity [4]. Real-time data integration further increases prediction accuracy. Our model-building process is iterative, which means it is constantly improved.

- *Logistic Regression (LR):*

Logistic Regression is a versatile linear model employed for the optimization of machine downtime. It models the probability of machine failures directly, offering a straightforward interpretation of the impact of various factors on downtime.

- *Naive Bayes:*

In the realm of machine downtime optimization, Naive Bayes provides a probabilistic approach, assuming independence between features. This model is particularly effective in scenarios where there's a need to analyse and predict the likelihood of downtime events based on various factors.

- *K-Nearest Neighbors (KNN):*

K-Nearest Neighbors is a valuable tool for machine downtime optimization, leveraging the similarity of instances to predict potential issues. By considering the characteristics of neighbouring instances, KNN contributes to identifying patterns and anomalies that may lead to downtime [23].

- *Decision Tree:*

Decision Trees are utilised in machine downtime optimization to hierarchically analyse factors contributing to downtime events. By recursively splitting the data based on different features, Decision Trees provide insights into the critical attributes influencing the system's reliability.

- *AdaBoost Classifier:*

AdaBoost stands out as a crucial ensemble method. Its effectiveness stems from the fusion of multiple weak learners, enabling it to adapt dynamically to the complexities inherent in the dataset. Through a process of iterative refinement, AdaBoost markedly improves the overall precision of downtime predictions.

- *Gradient Boosting:*

Gradient boosting works by gradually merging weak learners, typically decision trees, into one another [24]. It adjusts to the complexities of the dataset by highlighting the areas in which the model has previously performed poorly, allowing it to adapt to the intricacies of the dataset. Its versatility makes it excellent at identifying and logging the complex aspects and trends that lead to machine downtime. The result is a robust predictive model that can more efficiently optimise machine downtime by producing projections that are more reliable and accurate.

- *Bagging:*

Machine downtime optimization uses bootstrap aggregation, also known as bagging, to minimise bias and variation [25]. Bagging makes forecasts of possible downtime events more robust and dependable by creating several training sets and combining predictions from various models.

- *Random Forest:*

To maximise machine downtime, Random Forest, an ensemble learning technique, is constructed using many decision trees. Randomization in node and attribute selection enhances the model's capacity to capture complex links and improves downtime projections.

- *Extra Tree Classifier:*

Similar to Random Forest, the Extra Tree Classifier is used to optimise machine downtime. Through randomising node splits and utilising the complete dataset for training, Extra Trees helps reduce biases and variances, improving the model's performance.

- *Hist Gradient Boosting:*

Hist Gradient Boosting is instrumental in optimising machine downtime, particularly for handling large datasets efficiently. Its ability to iteratively refine predictions based on historical gradients makes it a robust choice for capturing patterns leading to downtime events.

F. Performance metric:

The efficacy of the models in detecting machine downtime was assessed through the application of several metrics, including F1 score, accuracy, sensitivity (recall), specificity, precision, and the Area Under the Receiver Operating Characteristic (AUROC) curve. The performance of the classifiers in differentiating between classes is comprehensively evaluated by these metrics [26]. An equivalent representation to a matrix called a confusion matrix, is used to compare the predicted class and the actual class during the evaluation phase [6,8]. From the confusion matrix [Fig.6], the necessary metrics listed below are derived.

- **Predicted Downtime and Actual Downtime (True Positive, TP):** Instances where the algorithm correctly forecasts machine downtime, and downtime does occur.
- **Predicted Normal Operation and Actual Normal Operation (True Negative, TN):** Instances where the algorithm accurately predicts normal machine operation and the machine indeed operates without downtime.
- **Predicted Downtime but Actual Normal Operation (False Positive, FP):** Instances where the algorithm incorrectly forecasts downtime, but the machine operates normally.
- **Predicted Normal Operation but Actual Downtime (False Negative, FN):** Instances where the algorithm erroneously predicts normal operation, but downtime occurs.

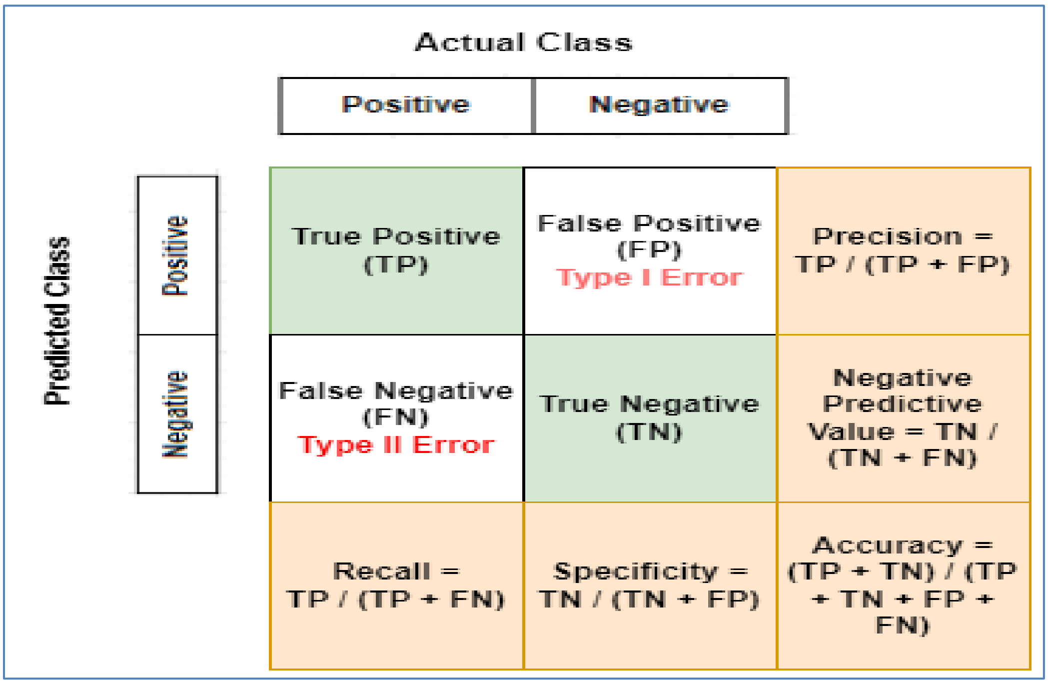

Fig. 6: Confusion matrix

- **F1-score:** The F1-score which is represented as

$$F1 = (2 \times \text{Precision} \times \text{Recall}) / (\text{Precision} + \text{Recall})$$

The f1-score, which ranges from 0 to 1 with higher values indicating a better balance between precision and recall [8], is especially helpful when both false positives missing actual downtime and false negatives predicting downtime when none exists. Aim for a high f1-score to make sure your model accurately identifies and predicts machine downtime while minimising false alarms.

- **Area Under Receiver Operating Characteristic (AUROC):** AUROC, an evaluation metric, is constructed by manipulating false positive and true positive rates [26]. It is a value ranging from 0 to 1, where a value closer to 1 indicates a good model. The AUROC curve provides insights into the trade-offs between sensitivity and specificity, offering a comprehensive view of the model's discriminatory abilities across different probability thresholds.

G. Model Building and Evaluation:

We optimised machine downtime using a diverse set of models, including Logistic Regression, Naive Bayes, KNN, Decision Tree, AdaBoost Classifier, Gradient Boosting, Bagging, Random Forest, Extra Tree Classifier, and Hist Gradient Boosting. To gauge their performance, we employed five evaluation metrics: accuracy, precision, recall, F1-score, and AUROC [Table 1].

Table 1: Performance analysis of the test models

Model Name	Accuracy	Precision	Recall	F1-Score	AUROC
Logistic regression	0.90	0.89	0.90	0.90	0.81
Naive Bayes	0.90	0.81	0.90	0.85	0.90
KNN	0.98	0.99	0.97	0.99	0.93
Decision Tree	0.97	0.94	0.95	0.94	0.92

ISSN No:-2456-2165

AdaBoost Classifier	0.93	0.97	0.95	0.98	0.94
Gradient Boosting	0.93	0.95	0.94	0.94	0.94
Bagging	0.91	0.94	0.93	0.94	0.96
Random Forest	0.98	0.95	0.96	0.96	0.95
Extra Tree Classifier	0.96	0.93	0.94	0.93	0.91
HistGradient Boosting	0.95	0.91	0.92	0.91	0.89

H. Hyper parameters:

For the machine downtime, it is imperative that the hyperparameters of each model be optimised. The model's capacity to generalise and generate trustworthy predictions depends on the careful selection and assessment of crucial parameters, including learning rates, tree depths, and regularisation strengths [27]. A methodical approach that includes a thorough analysis and testing of these hyperparameters [Table 2] is necessary to achieve the best results when optimising machine downtime.

Table 2: Model-Specific Hyperparameter Details

Model	Hyperparameters
Logistic Regression	- Penalty: ('l1', 'l2', 'elastic net', 'none') - C: Inverse of regularisation strength
Naive Bayes	- Type: Gaussian, Multinomial, or Bernoulli
K-Nearest Neighbors	- n_neighbors: Number of neighbors - weights: Weight function ('uniform' or 'distance') - p: Power parameter for Minkowski distance
Decision Tree	- Criterion: Measure of split quality ('gini' or 'entropy') - max_depth: Maximum depth of the tree - min_samples_split: Minimum samples to split an internal node
AdaBoost Classifier	- n_estimators: Number of boosting stages - learning_rate: Weight applied to each classifier
Gradient Boosting	- n_estimators: Number of boosting stages - learning_rate: Weight applied to each tree - max_depth: Maximum depth of individual trees
Bagging	- n_estimators: Number of base estimators in the ensemble
Random Forest	- n_estimators: Number of trees in the forest - criterion: Measure of split quality ('gini' or 'entropy') - max_depth: Maximum depth of the trees
Extra Tree Classifier	- n_estimators: Number of trees in the forest - criterion: Measure of split quality ('gini' or 'entropy') - max_depth: Maximum depth of the trees
HistGradient Boosting	- learning_rate: Weight applied to each tree - max_iter: Maximum number of iterations - max_depth: Maximum depth of the trees

In the realm of optimising machine downtime, an array of machine learning models was rigorously evaluated using key performance metrics, including accuracy, precision, recall, F1-score, and AUROC. Each model's distinctive strengths and trade-offs were meticulously examined to gauge their efficacy in predicting downtime events. Logistic Regression emerged with a commendable all-around performance, boasting a strong accuracy of 0.90, coupled with robust precision, recall, and F1-score metrics. Naive Bayes exhibited a balanced performance, particularly noteworthy for its accuracy of 0.90 and a well-balanced trade-off between precision and recall, as reflected in its F1-score of 0.85. K-Nearest Neighbors (KNN) demonstrated superior predictive accuracy, achieving an impressive accuracy of 0.98, indicating its robustness in downtime prediction. Decision Tree and Random Forest models excelled in overall metrics, showcasing high accuracy, precision, recall, and F1-score values.

As the selection process unfolds, the suitability of a model depends on the specific optimization goals. Considering a balanced performance across various metrics, Random Forest emerges as a strong candidate, boasting excellent accuracy and precision. The ensemble approach of Random Forest, leveraging multiple decision trees, contributes to its robust predictive capabilities. However, the final model choice should align with the nuanced priorities of the downtime optimization task, such as the importance of minimising false positives or maximising overall accuracy. Fine-tuning and potential ensemble techniques can further enhance model performance, providing a tailored solution to meet the unique challenges posed by machine downtime prediction.

III. HARDWARE SPECIFICATIONS

Machine downtime prediction depends on a number of factors, including the size of the dataset, the complexity of the model, and the desired prediction speed, which all affect the hardware specifications needed to run machine learning models [Table 3]. For machine downtime optimization, the following general recommendations have been specially designed:

Table 3: Optimal Hardware Specifications

Hardware Component	Recommended Specifications
RAM (Memory)	8 GB
CPU	Quad-core or higher for parallel processing
Storage	Adequate storage, SSD preferred for speed
Processor	Intel Core i5

IV. RESULTS AND DISCUSSION

Following rigorous model selection, the Random Forest model was refined through meticulous hyperparameter tuning, specifically targeting parameters to boost accuracy and minimise misclassification rates. This optimised model was seamlessly deployed in Streamlit, offering an intuitive platform for users to predict machine downtime events interactively [Fig.7]. The fine-tuned configuration not only elevated prediction accuracy but also reduced errors, ensuring reliable real-time insights within the Streamlit application. This streamlined integration underscores the model's readiness for practical deployment, aligning closely with the goal of minimising false positives and maximising overall accuracy in machine downtime prediction scenarios.

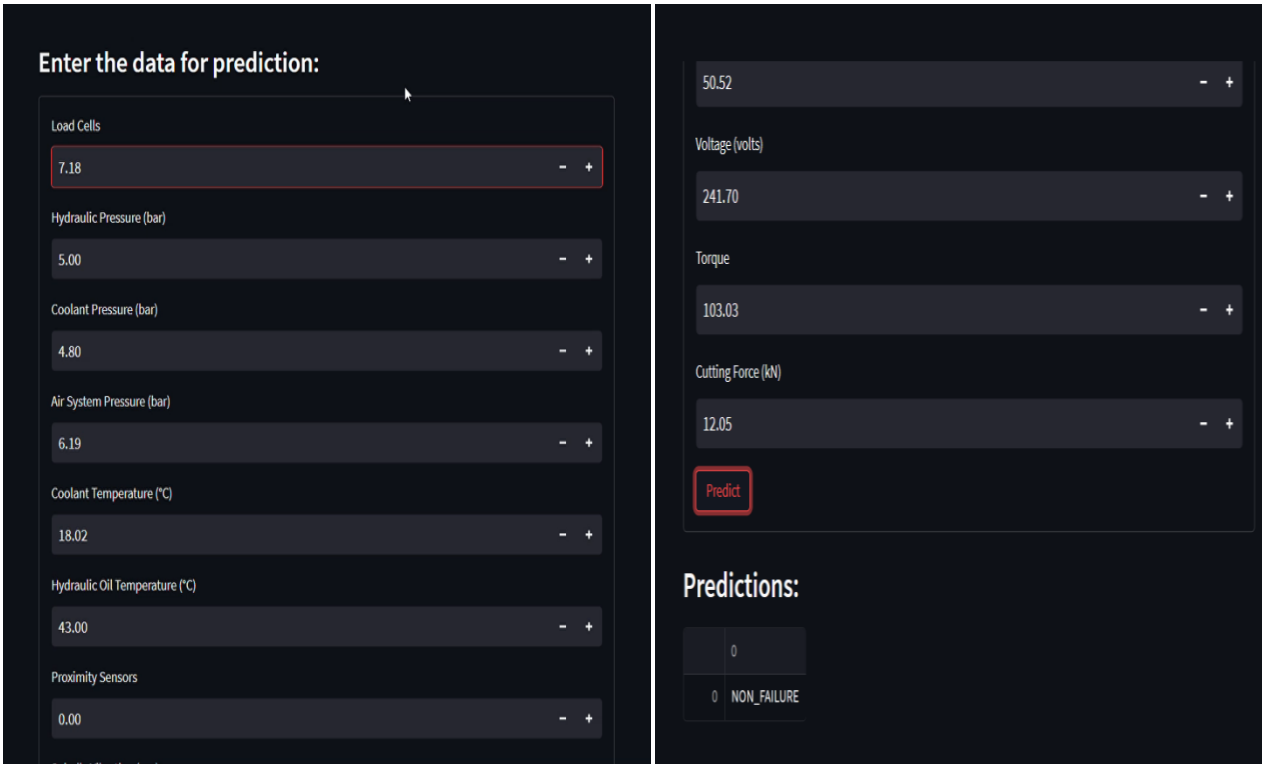

Fig. 7: Machine Downtime Prediction: A Streamlit Implementation

Ongoing enhancements will concentrate on refining the Random Forest model through continuous hyperparameter exploration for heightened accuracy. Exploring advanced ensemble techniques and incorporating user feedback will be crucial for adapting the model to changing operational dynamics. Future iterations aim to strengthen the model's resilience and relevance, ensuring it remains a reliable tool for dynamic machine downtime prediction scenarios.

V. CONCLUSION

Diverse performances were emphasised by the evaluation of the machine learning model for optimising machine downtime. Logistic Regression demonstrated commendable all-around performance, while Naive Bayes exhibited a balanced trade-off between precision and recall. K-Nearest Neighbors (KNN) demonstrated superior predictive accuracy, and Decision Tree and Random Forest models excelled in overall metrics.

The selection of the most suitable model depends on specific optimization goals. Random Forest emerged as a strong candidate due to its excellent accuracy and precision, leveraging an ensemble approach with multiple decision trees for robust predictive capabilities. However, the final choice should align with the nuanced priorities of the downtime optimization task, considering factors such as minimising false positives and maximising overall accuracy.

The importance of hyperparameter tuning was emphasised, as refining parameters such as learning rates, tree depths, and regularisation strengths play a pivotal role in enhancing model effectiveness. Fine-tuning and potential ensemble techniques were recommended to further improve model performance and provide a tailored solution for the unique challenges posed by machine downtime prediction.

In summary, the evaluation process underscored the need for a thoughtful analysis of model strengths, weaknesses, and hyper parameters to achieve optimal outcomes in the realm of optimising machine downtime. The findings provide valuable insights for practitioners seeking to implement effective machine-learning solutions for downtime prediction in industrial settings.

REFERENCES

[1]. Archit P. Kane, Ashutosh S. Kore, Advait N. Khandale, Sarish S. Nigade, Pranjali P. Joshi; Predictive Maintenance using Machine Learning. https://arxiv.org/abs/2205.09402
[2]. Jarosław Brodny and Magdalena Tutak; Applying Sensor-Based Information Systems to Identify Unplanned Downtime in Mining Machinery Operation. https://www.mdpi.com/1424-8220/22/6/2127
[3]. Radosław Wolniak; Downtime in the Automotive Industry Production Process – Cause Analysis.
[4]. https://www.qip-journal.eu/index.php/QIP/article/view/1259
[5]. Hamiche Ait Mimoune, Amine Boudghene Stambouli, Samir Flazi; Prioritising Machine Downtime and Simulation-Based Optimization System. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4057877
[6]. Stefan Studer, Thanh Binh Bui , Christian Drescher, Alexander Hanuschkin, Ludwig Winkler, Steven Peters, llandKlaus-Robert Müller; Towards CRISP-ML(Q): A Machine Learning Process Model with Quality Assurance Methodology. https://www.mdpi.com/2504-4990/3/2/20
[7]. Nadeem Iftikhar, Thorkil Baattrup-Andersen, Finn Ebertsen Nordbjerg, Eugen Bobolea1 and hhhPaul-Bogdan Radu; Data Analytics for Smart Manufacturing: A Case Study. https://www.researchgate.net/profile/Nadeem-Iftikhar/publication/333619125_Data_Analytics_for_Smart_Manufacturing_A_Case_Study/links/5e981ceba6fdcca7891c4457/Data-Analytics-for-Smaiiurt-Manufacturing-A-Case-Study.pdf
[8]. T. Liskiewicz, I. Sherrington, T. Khan , Y. Liu; Advances in sensing for real-time monitoring of tribological parameters. https://www.sciencedirect.com/science/article/pii/S0301679X23007533
[9]. Muhammad Zarar, Yulin Wang; Early Stage Diabetes Prediction by Approach Using Machine Learning Techniques. https://www.researchsquare.com/article/rs-3145599/v1
[10]. Tuga Mauritius, Annisa Safira Braza , Fransisca; Bank Marketing Data Mining using CRISP-DM Approach. https://www.warse.org/IJATCSE/static/pdf/file/ijatcse71852019.pdf
[11]. Deshalin Naidoo; Timothy T Adeliyi; Analysing University at-Risk Students in a Virtual Learning Environment using Machine Learning Algorithms. https://ieeexplore.ieee.org/abstract/document/10082752
[12]. Vasudevan K, Rajkumar, Karthick N, Suresh B, Prince Immanuel S; AUTOMATIC STEEL ROD CUTTING MACHINE BASED ON PLC AND SCADA. https://ijcrt.org/papers/IJCRT1893249.pdf
[13]. Mohd Javaid, Abid Haleem, Ravi Pratap Singh, Shanay Rab, Rajiv Suman; Significance of sensors for industry 4.0: Roles, capabilities, and applications. https://www.sciencedirect.com/science/article/pii/S2666351121000310
[14]. Nikhita Nadgauda, Senthil Arumugam Muthukumaraswamy; Design and Development of Industrial Automated System using PLC-SCADA. https://ieeexplore.ieee.org/abstract/document/9087620/authors#authors
[15]. Meir Kalech; Cyber-attack detection in SCADA systems using temporal pattern recognition techniques. https://www.sciencedirect.com/science/article/abs/pii/S0167404818307594
[16]. Julien Siebert, Lisa Joeckel, Jens Heidrich, Adam Trendowicz, Koji Nakamichi, Kyoko Ohashi, Isao Namba, Rieko Yamamoto & Mikio Aoyama; Construction of a quality model for machine learning systems. https://link.springer.com/article/10.1007/s11219-021-09557-y
[17]. Stefan Studer, Thanh Binh Bui, Christian Drescher, Alexander Hanuschkin, Ludwig Winkler, Steven Peters, Klaus-Robert Müller; Towards CRISP-ML(Q): A Machine Learning Process Model with Quality Assurance Methodology. https://arxiv.org/pdf/2003.05155.pdf
[18]. Peter Sugimura, Florian Hartl; Building a Reproducible Machine Learning Pipeline. https://arxiv.org/ftp/arxiv/papers/1810/1810.04570.pdf
[19]. Kabita Sahoo, Abhaya Kumar Samal, Jitendra Pramanik, Subhendu Kumar Pani; Exploratory Data Analysis using Python. https://www.researchgate.net/profile/Dr-Subhendu-Pani/publication/337146539_IJITEE/links/5dc70b124585151435fb427e/IJITEE.pdf
[20]. Chris Chatfield; Exploratory data analysis. https://www.sciencedirect.com/science/article/abs/pii/0377221786902092

[21]. Noviyanti T M Sagala, Fonggi Yudi Aryatama; Exploratory Data Analysis (EDA): A Study of Olympic Medallist. http://sistemasi.ftik.unisi.ac.id/index.php/stmsi/article/view/1857
[22]. Hall, Mark A.; Correlation-based feature selection for machine learning. https://researchcommons.waikato.ac.nz/handle/10289/15043
[23]. Michael Onyema Edeh, Osamah Ibrahim Khalaf, Carlos Andrés Tavera, Sofiane Tayeb, Samir Ghouali,
[24]. Ghaida Muttashar Abdulsahib, Nneka Ernestina Richard-Nnabu, AbdRahmane Louni; A Classification Algorithm-Based Hybrid Diabetes Prediction Model. https://www.frontiersin.org/articles/10.3389/fpubh.2022.829519/full
[25]. Nurjahan Nipa, Mahmudul Hasan Riyad, Shahriare Satu, Walliullah, Koushik Chandra Howlader, Mohammad Ali Moni; Clinically adaptable machine learning model to identify early appreciable features of diabetes in Bangladesh. https://www.sciencedirect.com/science/article/pii/S2667102623000049
[26]. Jerome H. Friedman; Greedy Function Approximation: A Gradient Boosting Machine. https://www.jstor.org/stable/2699986
[27]. Dhyan Chandra Yadav & Saurabh Pal; An experimental study of diversity of diabetes disease features by bagging and boosting ensemble method with rule based machine learning classifier algorithms. https://link.springer.com/article/10.1007/s42979-020-00446-y
[28]. Md. Kowsher, Anik Tahabilder, Nusrat Jahan Prottasha, Kaiser Habib; Predicting the Appropriate Mode of Childbirth using Machine Learning Algorithm. https://www.researchgate.net/profile/Md-Kowsher-2/publication/352054610_Predicting_the_Appropriate_Mode_of_Childbirth_using_Machine_Learning_Algorithm/links/60b6ffcda6fdcc476bde7fbf/Predicting-the-Appropriate-Mode-of-Childbirth-using-Machine-Learning-Algorithm.pdf
[29]. Patrick Schratz, Jannes Muenchow, Eugenia Iturritxa, Jakob Richter, Alexander Brenning; Hyperparameter tuning and performance assessment of statistical and machine-learning algorithms using spatial data. https://www.sciencedirect.com/science/article/abs/pii/S0304380019302145.