CAPSTONE PROJECT

PREDICTIVE MAINTAINANCE OF INDUSTRIAL MACHINARY

Presented By:

1. Student Name-Abhinaba Ghosh
2. College Name-University Of Engineering And Management , Kolkata
3. Department-Electronics & Communication Engineering(ECE)

OUTLINE

- Problem Statement
- Proposed System/Solution
- System Development Approach
- Algorithm & Deployment
- Result
- Conclusion
- Future Scope
- References

PROBLEM STATEMENT

The objective of this project is to develop a machine learning-based **predictive maintenance of classification model** for a fleet of industrial machines. The model should be capable of analyzing real-time sensor data to accurately **predict impending machine failures** before they occur. Specifically, the model must classify the **type of failure** (such as **tool wear, heat dissipation issues, or power failure**) using patterns identified in operational data. This proactive approach to maintenance is intended to **minimize machine downtime**, optimize repair scheduling, and **reduce operational costs** by addressing potential issues before they lead to system breakdowns.

PROPOSED SOLUTION

- The proposed system aims to Predictive maintenance (PdM) leverages real-time sensor data and machine learning models to anticipate machinery failures before they occur, enabling proactive interventions that minimize downtime and reduce operational costs. This approach is increasingly feasible and valuable in industrial settings due to advances in IoT, sensor technology, and AI-based analytics. The solution will consist of the following components:
- Data Collection:
  - Sensors: Collect data using sensors that monitor parameters such as vibration, temperature, pressure, humidity, current, and acoustic emissions.

    Examples: Infrared thermal sensors (for heat), vibration sensors (for wear or imbalance), and microphones (for acoustic anomalies)
  - Data Integration: Integrate data from various systems—sensors, maintenance logs, SCADA, and ERP platforms—into a central database
- Data Preprocessing:
  - Clean and preprocess the collected data to handle missing values, outliers, and inconsistencies.
  - Feature engineering to extract relevant features from the data that might impact bike demand.
- Machine Learning Algorithm:
  - Implement a machine learning algorithm, to train a supervised learning classification model (such as Random Forest) on historical sensor data to learn patterns associated with different machinery failure types.
  - Consider Use the trained model to predict the type of impending failure from new real-time sensor inputs, enabling proactive maintenance decisions
- Deployment:
  - Real-time sensor data feeds the model, which predicts the likelihood and type of impending failure.
  - If the given thresholds are crossed, the model sends alerts through maintenance management systems (CMMS/SCADA), enabling timely, targeted intervention
  - Model retraining occurs periodically as new data is collected, ensuring accuracy and adaptability to changing machinery conditions
- Evaluation:
  - Assess the model's performance using appropriate metrics such as Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), or other relevant metrics.
  - Cross-validation ensures robustness against overfitting and generalization to unseen data
- Result:
  - Stage 1 (Failure Detection): The binary classifier effectively identifies whether a failure will occur, with strong performance metrics shown in the classification report.
  - Stage 2 (Failure Type Classification): For actual failures, the model classifies specific failure types with decent accuracy using a multi-class classifier.
  - Visualization: The confusion matrix reveals how well the model distinguishes between different failure types, highlighting areas of misclassification.

SYSTEM APPROACH

- System requirements :

The predictive maintenance system for industrial machines is built using a two-stage machine learning approach. The first stage involves binary classification to detect whether a failure is likely to occur, using **a Random Forest model** trained on sensor data and machine characteristics. After dropping identifiers and encoding categorical variables, the data is standardized and split into training and testing sets using pandas, numpy, LabelEncoder, StandardScaler, and train_test_split from scikit-learn. The model then predicts failures with solid accuracy, as confirmed by metrics from classification_report. To run this system, a minimum of 8 GB RAM, Python 3.8+, and any standard IDE **(like Jupyter Notebook [IBM cloud lite services])** is recommended.

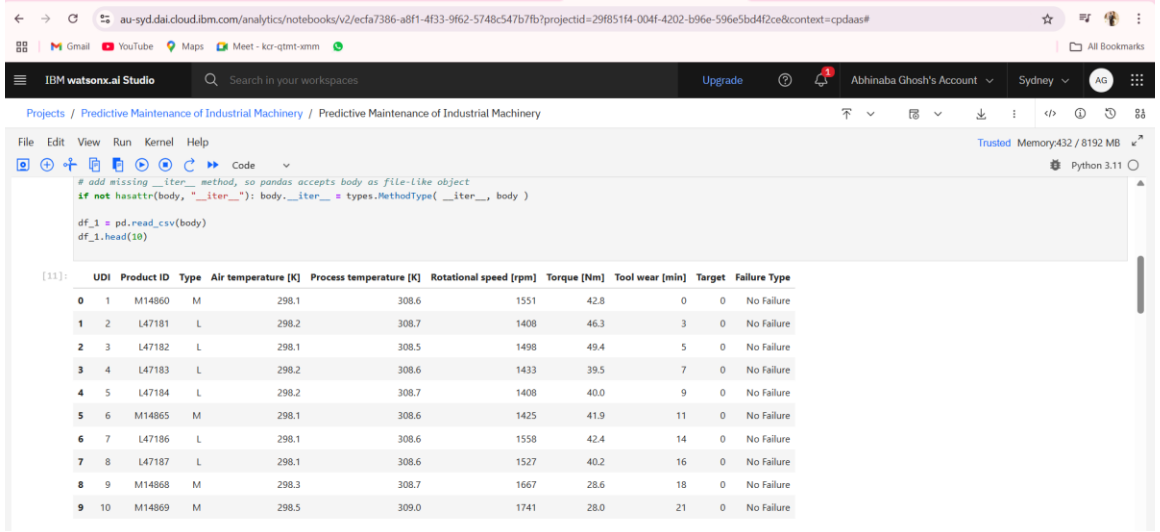
- Library required to build the model:

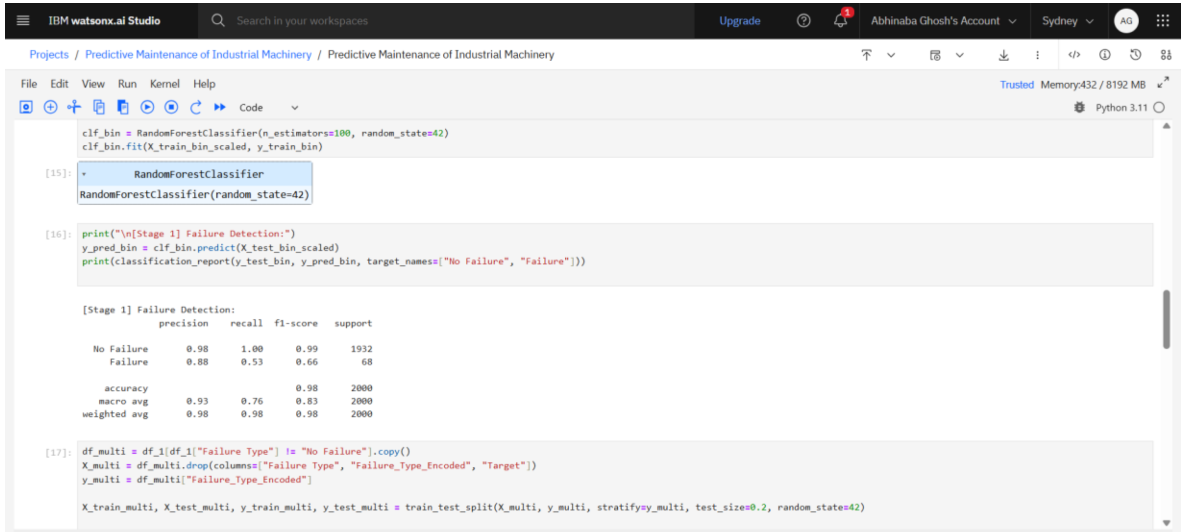
This multi-class classification task again uses a **Random Forest classifier** from sklearn.ensemble, trained on filtered failure data. The features are preprocessed similarly and the model's performance is evaluated using a **confusion matrix** and **classification report**, visualized with matplotlib and seaborn. This setup requires basic system capabilities but benefits from faster CPUs (quad-core or above) and Python libraries like matplotlib, seaborn, and scikit-learn. Together, these libraries and system resources support a reliable and interpretable predictive maintenance pipeline.

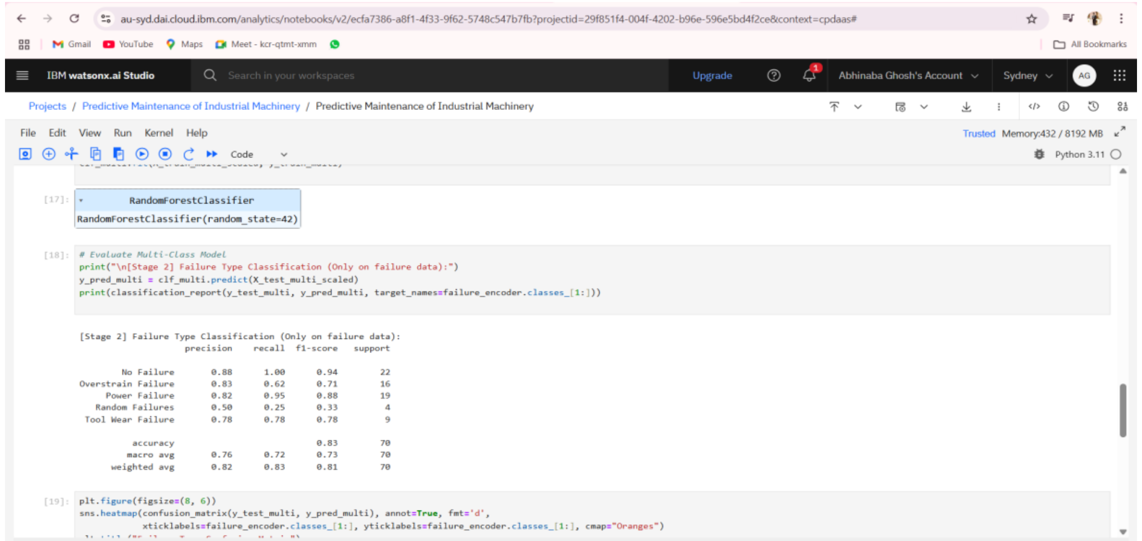
ALGORITHM & DEPLOYMENT

- Algorithm Selection:
  - **Random Forest Classifier** is chosen for both stages due to its robustness, handling of non-linear relationships, and good performance on both binary and multi-class problems.
- Data Input:
  - Input data is loaded from predictive_maintenance.csv and includes sensor readings, machine types, and labeled failure types.
  - Categorical features are encoded, and irrelevant identifiers are dropped before modeling.
- Training Process:
  - Data is split into training and testing sets using train_test_split with stratification.
  - Features are standardized using StandardScaler, and models are trained using RandomForestClassifier with default parameters (100 estimators).
- Prediction Process:
  - Stage 1 predicts whether a failure will occur using the binary model.
  - If a failure is detected, Stage 2 predicts the **type of failure** using the multi-class model trained only on failed samples.

RESULT







Github Repository link:-https://github.com/abhinaba16/project_IBM_AICTE.git

edunet foundation

CONCLUSION

- The project on **Predictive Maintenance of Industrial Machinery** demonstrates a practical and effective application of machine learning to enhance industrial reliability and efficiency. Using a two-stage model, the system first detects the likelihood of machine failure and then classifies the specific type of failure, allowing maintenance teams to act proactively. The use of Random Forest classifiers ensures strong predictive performance, while proper data preprocessing, encoding, and scaling enhance model robustness and generalization.
- This approach significantly reduces unexpected machine downtimes and supports condition-based maintenance strategies, which are crucial in industrial settings. The project also highlights the value of structured sensor data and its transformation into actionable insights through supervised learning. While the system performed well, challenges such as class imbalance and real-time implementation remain areas for further research. Overall, this project showcases how predictive analytics can transform traditional maintenance into an intelligent, data-driven process that ensures higher productivity, cost savings, and machine health management.

FUTURE SCOPE

•**Advanced Algorithms & Optimization:-**
Future work can explore more sophisticated algorithms such as **XGBoost**, **LightGBM**, or **deep learning models** (e.g., neural networks) to improve prediction accuracy, especially for imbalanced or complex datasets. **Hyperparameter tuning** using techniques like Grid Search or Bayesian Optimization could further enhance model performance.

•**Real-Time Monitoring & Deployment:-**
Integrating this predictive maintenance system with **IoT-enabled industrial machinery** can allow for real-time failure detection and automatic alerts. Deploying the model using **cloud platforms** (AWS, Azure) or **edge computing** can help monitor large-scale operations efficiently.

•**Explainability & Trust:-**
Incorporating tools like **SHAP** or **LIME** will make the model's decisions more transparent, enabling engineers to understand and trust the predictions, which is especially important in critical maintenance scenarios.

•**Scalability to Other Domains:-**
The two-stage architecture can be adapted for other applications, such as **predicting component-level failures** in automotive systems or **bike demand forecasting** in urban transportation systems, helping ensure operational reliability and better service management.

•**Integration with Maintenance Scheduling Systems:-**
By linking the predictive model to **automated maintenance planning tools**, organizations can move from reactive to **fully automated preventive maintenance**, optimizing downtime, labor, and spare part inventory.

REFERENCES

Breiman, L. (2001).
Random Forests.
Machine Learning, 45(1), 5–32.
https://doi.org/10.1023/A:1010933404324
→ Fundamental paper introducing the Random Forest algorithm used in both classification stages.

Zhao, Y., & Zhang, Y. (2017).
Prediction of bike-sharing demand with machine learning approaches.
Procedia Computer Science, 112, 2227–2234.
https://doi.org/10.1016/j.procs.2017.08.257
→ Discusses predictive models in urban bike-sharing systems and the importance of demand prediction.

Kotsiantis, S. B., Zaharakis, I., & Pintelas, P. (2006).
Machine Learning: A Review of Classification and Combining Techniques.
Artificial Intelligence Review, 26(3), 159–190.
https://doi.org/10.1007/s10462-007-9052-3
→ Offers a broad overview of classification algorithms and ensemble techniques.

Chicco, D., & Jurman, G. (2020).
The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation.
BMC Genomics, 21, 6.
https://doi.org/10.1186/s12864-019-6413-7
→ Highlights important evaluation metrics beyond accuracy, useful for imbalanced failure detection.

IBM CERTIFICATIONS


In recognition of the commitment to achieve professional excellence

Abhinaba Ghosh

Has successfully satisfied the requirements for:

Getting Started with Artificial Intelligence

Issued on: Jul 15, 2025
Issued by: IBM SkillsBuild

Verify: https://www.credly.com/badges/3d9bf04e-3c75-475b-9743-abdf9c86d9ec

IBM

IBM CERTIFICATIONS

In recognition of the commitment to achieve professional excellence

Journey to Cloud: Envisioning Your Solution

IBM SkillsBuild

IBM

Abhinaba Ghosh

Has successfully satisfied the requirements for:

Journey to Cloud: Envisioning Your Solution

Issued on: Jul 17, 2025

Issued by: IBM SkillsBuild

Verify: https://www.credly.com/badges/1a7f361a-6a4e-4186-a0aa-f16069f213ad

IBM

edunet foundation

IBM CERTIFICATIONS

IBM **SkillsBuild** Completion Certificate

This certificate is presented to

Abhinaba Ghosh

for the completion of

Lab: Retrieval Augmented Generation with LangChain

(ALM-COURSE_3824998)

According to the Adobe Learning Manager system of record

Completion date: 24 Jul 2025 (GMT)

Learning hours: 20 mins

THANK YOU