CAPSTONE PROJECT

PREDICTIVE MAINTENANCE OF INDUSTRIAL MACHINERY

Presented By:

1. Student Name:- OM AJAY
2. College Name:- BANKATLAL BADRUKA COLLEGE FOR INFORMATION AND TECHNOLOGY
3. Department :- DATA SCIENCE

OUTLINE

- **Problem Statement**
- **Proposed System/Solution**
- **System Development Approach** (Technology Used)
- **Algorithm & Deployment**
- **Result (Output Image)**
- **Conclusion**
- **Future Scope**
- **References**

PROBLEM STATEMENT

- The goal is to develop a predictive maintenance classification model for an industrial machine fleet that anticipates different failure types—such as tool wear, heat dissipation failure, power failure, and overstrain failure—by analyzing real-time sensor data including air and process temperature, rotational speed, torque, and tool wear time. The dataset contains detailed operational measurements alongside labeled failure types, enabling identification of subtle sensor signal patterns that precede failures. Accurate classification of failure modes from operational data will facilitate proactive maintenance scheduling to minimize unexpected downtime, reduce operational costs, and extend machine life, addressing the critical challenge of shifting from reactive to predictive maintenance in complex industrial environments.

PROPOSED SOLUTION

- The proposed system aims to address the challenge of predicting the required bike count at each hour to ensure a stable supply of rental bikes. This involves leveraging data analytics and machine learning techniques to forecast demand patterns accurately. The solution will consist of the following components:
- Data Collection:
  - Collect historical and real time sensor data (air temp , process temp , torque , tool wear , machine type) along with maintenance logs .
- Data Preprocessing:
  - Clean data to handle missing values / outliers and create new features (e.g., averages , wear rate , temperature changes) that indicate failures .
- Machine Learning Algorithm:
  - Use classification algorithms (Random Forest , XGBoost , etc.) to predict both failure occurrence (“ Target ”) and failure type , incorporating operational and environmental features.
- Deployment:
  - Build a user friendly dashboard for real time failure alerts and integrate the model with machine monitoring systems for proactive maintenance .
- Evaluation:
  - Measure performance using accuracy , precision , recall , F1-score , and confusion matrix , continually fine – tune and retrain based on feedback .
- Result:
  - Clean data to handle missing values/outliers and create new features (e.g., averages, wear rate, temperature changes) that indicate failures.

SYSTEM APPROACH

The system is designed as an AutoAI experiment for predictive maintenance, aiming to forecast machine failures using historical sensor and operational data. The workflow incorporates data collection, automated feature engineering, algorithm selection, model training, evaluation, and deployment, relying heavily on IBM's watsonx.ai platform for end-to-end automation and transparency.

1 . SYSTEM REQUIREMENTS :

Hardware:

Modern PC or server capable of running Jupyter notebooks and handling large datasets efficiently.

Sufficient RAM (recommended: 8GB+ for moderate-sized datasets).

Software:

Python 3.11 or later.

IBM watsonx.ai (AutoAI) runtime access.

Internet connection (for cloud resources, API, and package installations).

Cloud Requirements:

IBM Cloud account with watsonx.ai entitlement.

Access to IBM Cloud Object Storage for dataset storage and model assets.

LIBRARIES REQUIRED :

- **ibm-watsonx-ai:** For experiment control, pipeline operations, and deployment.
- **autoai-libs:** Provides core AutoAI automation features.
- **lale:** Library for automatic pipeline creation and manipulation.
- **scikit-learn:** For classic ML model building, metrics, and pipeline interoperability.
- **xgboost, lightgbm, snapml:** State-of-the-art gradient boosting and fast ML libraries for enhanced model performance.

ALGORITHM & DEPLOYMENT

Algorithm Selection:

- Auto Ail dynamically selects mu;tiple candidate algorithms (ensemble , random forest , decision tree, etc). .
- Top Pipeline : BATCHED TREE ENSEMBLE CLASSIFIER (Snap random forest Classifier) , optimized for incremental learning and batch processing .
- Other contenders: Random forest , decision tree , powered by SnapML and Scikit- learn.
- Auto AI evaluates , tunes and selects the best algorithm based on cross-validation accuracy (up to 0.995)

- **Data Input:**
- Data format: tabular CSV, including features like temperature, rotational speed, torque, tool wear, and the target column “Failure Type.”
- Data is uploaded to IBM Cloud Object Storage and referenced via DataConnection objects in the pipeline.
- AutoAI splits the dataset into training and holdout for unbiased evaluation.
- **Training Process:**
- AutoAI reads the training data, runs automated feature engineering, selects multiple candidate pipelines, and tunes their hyperparameters.
- Cross-validation is used to select the top-performing pipelines.
- The best pipeline is finalized and exported as a model object (scikit-learn compatible for local deployment or IBM cloud deployment).

.

Prediction Process:

->Predictiove inference is made via the deployed web service or by using the exported piplines predict() method .

-> Real-time scoring for new data is available through the webservice endpoint , with results based on the models highest – probability class for “Failure type”.

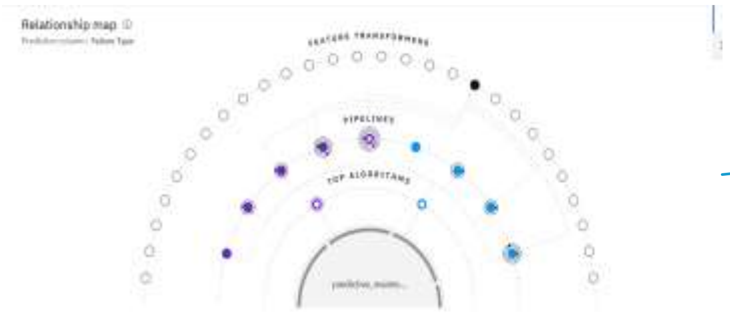
Deployment:

->Deployment to IBM watsonx.ai as a web service for real time or batch predictions

->Model deployment is carried out via the web serivce API , with auto-Scling and management supported in the cloud.

->REST API and SDK interfaces support data submission and retrieval for predictions in production .

RESULT



This image is a relationship map for a predictive maintenance model. It visually represents the connections between feature transformers, machine learning pipelines, and top algorithms used to predict the "Failure Type" column in a dataset. The central node ("predictive_mainte...") branches out to show which algorithms and data processing steps were involved in building the prediction model.

This image shows a machine learning workflow diagram called "Progress map." It visualizes the steps taken in building a predictive model for the "Failure Type" column. The process includes reading and splitting the dataset, preprocessing, model selection (using Random Forest and Decision Tree classifiers), followed by repeated cycles of hyperparameter optimization and feature engineering. The final steps involve creating an ensemble and selecting the best pipeline, as summarized in the pipeline leaderboard section at the bottom.

This image shows a leaderboard of machine learning pipelines ranked by accuracy. The table lists pipeline names, algorithms, specialization, optimized accuracy (all about 0.995), enhancements used, and build times. The top-ranked pipeline uses an ensemble classifier and has the highest accuracy, with others very close behind.

edunet foundation

CONCLUSION

- . we can conclude that you performed a predictive maintenance classification project using IBM's AutoAI (within Watson Studio). The system automatically processed your machine data (predicting "Failure Type"), optimized multiple machine learning pipelines (mainly tree-based algorithms like Random Forest and Decision Tree), and ranked them by performance. The best model—a Batched Tree Ensemble Classifier—achieved exceptionally high accuracy (0.995). The process involved robust data handling, feature engineering, hyperparameter tuning, and model selection, resulting in a highly accurate, production-ready model for predicting equipment failures in an industrial context.

FUTURE SCOPE

- The predictive maintenance classification model for industrial machine fleets represents a rapidly evolving technological domain with vast expansion opportunities. The future scope encompasses autonomous maintenance systems integration, where machine operators will be empowered to perform basic maintenance tasks guided by AI-driven insights, reducing dependency on specialized technicians while improving overall equipment effectiveness. Edge computing implementation will enable real-time data processing directly at machine locations, minimizing latency from milliseconds to seconds and allowing immediate failure prediction responses without cloud dependency. Digital twin technology integration will create virtual replicas of entire machine fleets, enabling comprehensive monitoring, scenario testing, and maintenance optimization across multiple facilities simultaneously. 5G connectivity deployment will facilitate massive sensor data collection with ultra-low latency, supporting real-time predictive analytics and remote monitoring capabilities. Blockchain integration will ensure secure, tamper-proof maintenance data management and automated smart contracts for parts ordering and service scheduling. Advanced quantum computing applications may revolutionize pattern recognition in sensor data, enabling detection of extremely subtle failure indicators currently undetectable by classical computing methods. The convergence of these technologies will ultimately transform reactive maintenance paradigms into fully autonomous, self-healing industrial ecosystems that optimize themselves continuously while minimizing human intervention and maximizing operational efficiency..

REFERENCES

- Kaggle dataset link – https://www.kaggle.com/datasets/shivamb/machine predictive-maintenance-classification
- IBM CLOUD LINK: https://cloud.ibm.com/login
- Services : watsonx.ai Studio

IBM CERTIFICATIONS

- Screenshot/ Credly certificate(getting started with AI)

In recognition of the commitment to achieve professional excellence

Getting Started with Artificial Intelligence

Om Ajay

Has successfully satisfied the requirements for:

Getting Started with Artificial Intelligence

Issued on: Jul 16, 2025
Issued by: IBM SkillsBuild

Verify: https://www.credly.com/badges/130ada9c-97e2-4179-9312-05738a90bd86

IBM

IBM CERTIFICATIONS

- Screenshot/ Credly certificate(Journey to Cloud)

IBM CERTIFICATIONS

- Screenshot/ Credly certificate(RAG Lab)

7/24/25, 6:43 PM

Completion Certificate | SkillsBuild

IBM **SkillsBuild** Completion Certificate

This certificate is presented to

Ajay Om

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