CAPSTONE PROJECT

PREDICTIVE MAINTENANCE OF INDUSTRIAL MACHINERY

Presented By:

1. Swapnil Patidar-BMSCE-ISE Department

edunet foundation

OUTLINE

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

PROBLEM STATEMENT

- In industrial settings, the unexpected failure of machinery is a primary cause of operational disruption.
- These failures lead to costly, unplanned downtime, increased maintenance expenses, and significant reductions in production output.
- The current maintenance approach is often reactive, meaning action is taken only after a failure has already occurred.
- The crucial challenge is to anticipate these failures before they happen and to understand the specific type of fault that is developing.

PROPOSED SOLUTION

- The proposed system aims to develop a predictive maintenance model to anticipate and classify machine failures before they occur. The solution will consist of the following components:
- **Data Collection:**
  - Utilize the "Machine Predictive Maintenance Classification" dataset from Kaggle, containing sensor data from industrial machinery.
- **Data Preprocessing:**
  - Clean the collected sensor data to handle any inconsistencies or missing values.
  - Perform feature engineering to extract the most relevant operational parameters that influence machine failure.
- **Machine Learning Algorithm:**
  - Develop a multi-class classification model capable of predicting the specific type of failure (e.g., tool wear, heat dissipation, power failure, overstrain).
  - The model will be trained on historical sensor data to learn the patterns that precede a failure and predict what type of failure is going to occur.
  - **We will use Hyperparameter optimization to increase the accuracy of the model.**
- **Deployment:**
  - **Develop a user-friendly interface or application that provides real-time predictions for bike counts at different hours.**
  - **Deploy the solution on a scalable and reliable platform, considering factors like server infrastructure, response time, and user accessibility.**
- **Evaluation:**
  - Assess the model's performance using standard classification metrics such as Accuracy, Precision, Recall, and F1-Score.
  - **Fine-tune the model based on feedback and continuous monitoring of prediction accuracy.**

SYSTEM APPROACH

- **System requirements:**
  - **Language**: Python 3.11.
  - **Platform**: IBM Cloud Lite Services, specifically leveraging **IBM Watson Studio** with the **watsonx.ai** client for model development and deployment.
- **Library required to build the model:**
  - **Core: AI/ML**:
    - IBM watsonx.ai Studio (for platform interaction)
    - IBM watsonx ai Runtime service (for building the model)
    - IBM Cloud Object Storage (for storing and working on model and deployment)
    - autoai-libs, scikit-learn (for modelling and metrics)
    - snapml, and lale (for pipeline optimization)
  - **Data Visualization:**
    - matplotlib for plotting the learning curve.

ALGORITHM & DEPLOYMENT

- **Algorithm Selection:**
  - The model is an optimized pipeline (Pipeline_5) generated by IBM's AutoAI feature.
  - The core algorithm is a **Snap Random Forest Classifier**, a high-performance version of the standard Random Forest algorithm.
  - It's a multi-class classification model trained to predict the specific failure type from six possible classes: **'Heat Dissipation Failure', 'No Failure', 'Overstrain Failure', 'Power Failure', 'Random Failures', and 'Tool Wear Failure'**.
- **Data Input:**
  - The input features used by the algorithm are: **UDI, Product ID, Type, Air temperature [K], Process temperature [K], Rotational speed [rpm], Torque [Nm], Tool wear [min], Target to predict the type of failure that can occur.**
- **Training and Prediction Process:**
  - The model is trained using an incremental learning approach with the partial_fit method, which allows the model to continue learning as new data arrives without retraining from scratch.
  - Training data is processed in batches of 10,000 rows to simulate a real-world data stream.
- **Deployment:**
  - The final trained pipeline is stored as a model asset in the IBM WatsonX repository.
  - The model is deployed as a real-time online web service, which allows for making live predictions by sending new sensor data to an API endpoint.

RESULT

Pipeline 5	Batched Tree Ensemble Classifier (Snap Random Forest Classifier)	INCR	0.995	HPO-1	FE	HPO-2	BATCH

- The model achieved a training accuracy of 99.5%.
- The Prediction Results are:

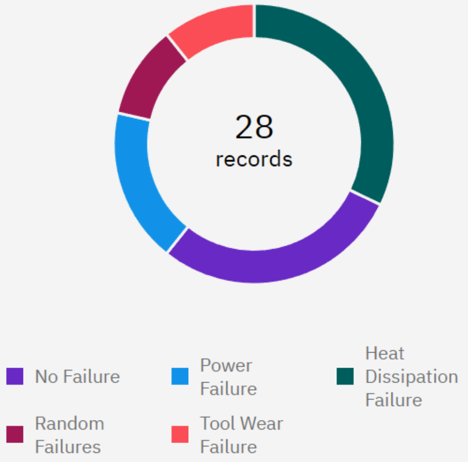
Prediction results

×

Prediction type

Multiclass classification

Prediction percentage

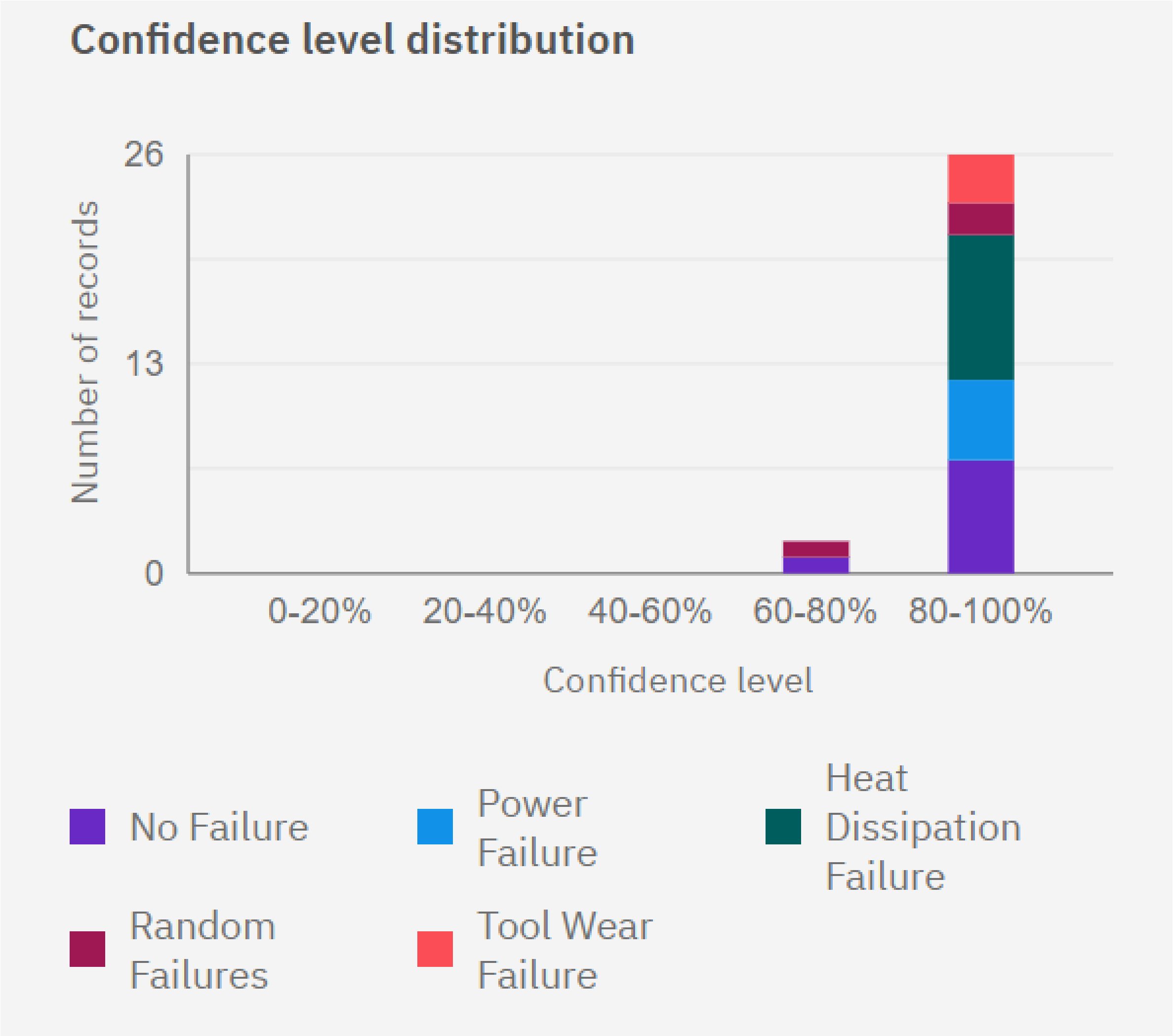


Display format for prediction results

Table view   JSON view   Show input data

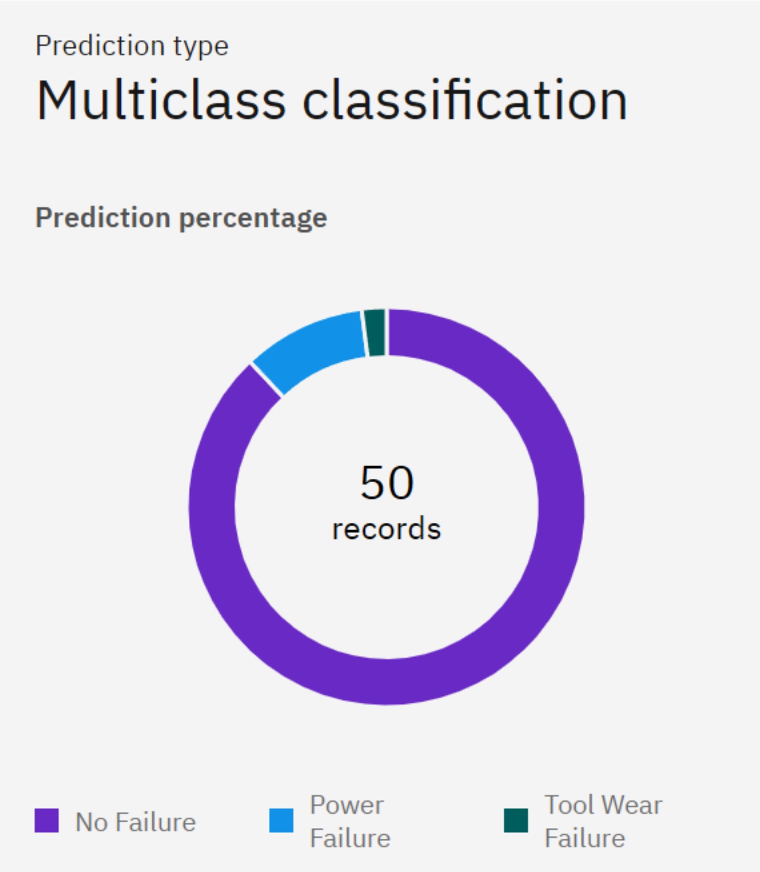
	Prediction	Confidence	UDI	Product ID	Type	Ai
1	No Failure	100%		H29581	H	29
2	Power Failure	100%	848	L48027	L	29
3	Power Failure	100%	208	M15067	M	29
4	Power Failure	100%	260	M15119	M	29
5	Power Failure	100%	381	L47560	L	29
6	Power Failure	90%	443	L47622	L	29
7	Heat Dissipation Failure	90%	3807	M18666	M	30
8	Heat Dissipation Failure	100%	3815	M18674	M	30
9	Heat Dissipation Failure	100%	3830	H33243	H	30
10	Heat Dissipation Failure	90%	4079	H33492	H	30
11	Heat Dissipation Failure	90%	3807	M18666	M	30
12	Heat Dissipation Failure	100%	3815	M18674	M	30

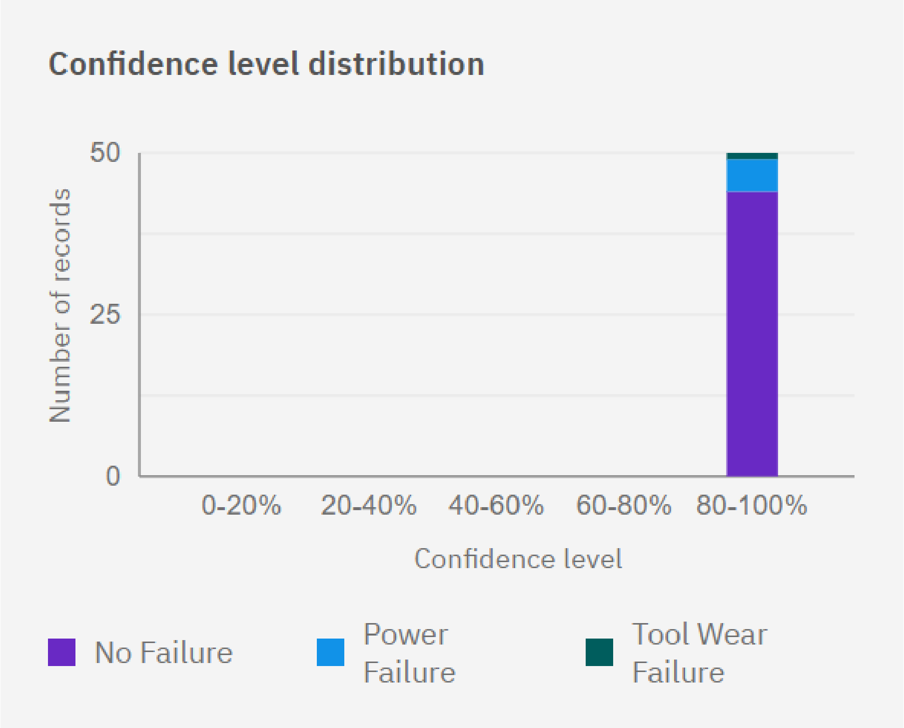
RESULT


Confidence level distribution
26
13
0
Number of records
0-20%
20-40%
40-60%
60-80%
80-100%
Confidence level
No Failure
Power Failure
Heat Dissipation Failure
Random Failures
Tool Wear Failure

RESULT

- This model predicted all the data with 100% accuracy with an average confidence score of 96%.
- The model was also able to accurately predict the type of failure on unknown synthetic data.





CONCLUSION

- This project successfully utilized IBM's AutoAI to select and build an efficient classifier (Snap Random Forest) for predicting industrial machinery failures.
- The use of incremental learning (partial_fit) demonstrates a practical approach for updating the model in dynamic environments where data is continuously generated.
- The model was successfully deployed as a scalable online web service on the IBM Cloud platform, making it ready for integration into a real-world predictive maintenance application.
- The outcome is a functional, deployable system that enables a proactive maintenance strategy, directly addressing the core problem of reducing costly downtime.

FUTURE SCOPE

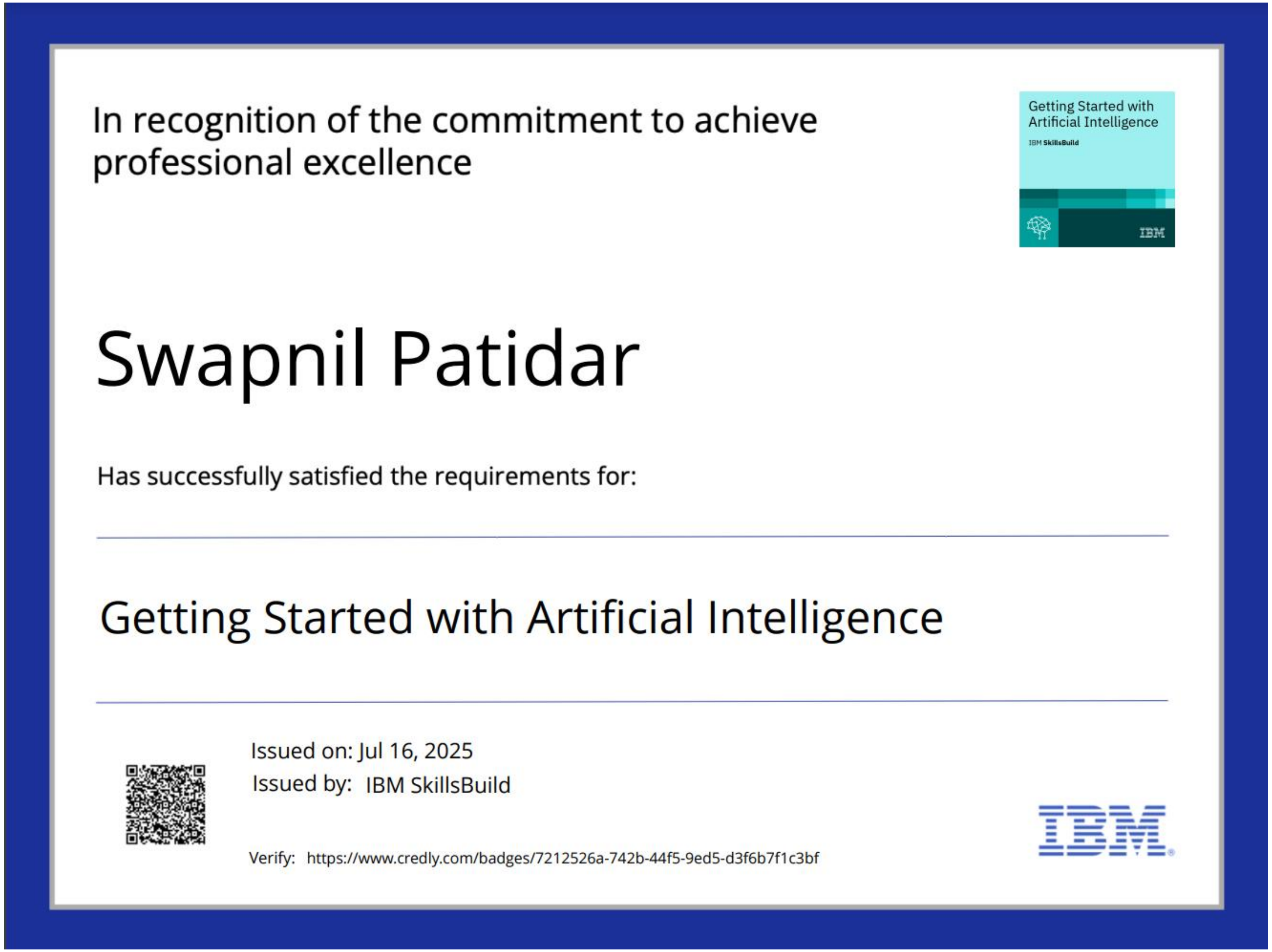
- Real-World Pilot: Integrate the deployed web service with live sensor data from a pilot group of machines to validate its real-world performance and business impact.
- Concept Drift Monitoring: Implement mechanisms to monitor the model's performance over time to detect "concept drift" (when real-world data patterns change) and trigger retraining.
- Predict Remaining Useful Life (RUL): Develop a companion regression model to predict the exact time remaining until a predicted failure occurs.
- Refine Incremental Strategy: Experiment with different batch sizes and sampling strategies to further optimize the incremental learning process.

REFERENCES

- Project & Dataset Sources:
  - Dataset:
    - Shivam, B. (2020). Machine Predictive Maintenance Classification. Kaggle. Retrieved from: https://www.kaggle.com/datasets/shivamb/machine-predictive-maintenance-classification
  - Problem Statement Source:
    - Edunet Foundation & IBM. (2024). Problem Statements on ML by using AI_KOSH data sets. SB4Academia_Problem Statements_2025 (1).pdf.
  - Notebook & Methodology:
    - IBM. (2025). P5 - Snap Random Forest Classifier_ Model Builder.ipynb. AutoAI Generated Notebook, IBM Watson Studio.
- Academic & Technical Literature:
  - On the Core Algorithm (Random Forest):
    - Breiman, L. (2001). Random Forests. Machine Learning, 45(1), 5-32.
    - This is the original, seminal paper by Leo Breiman that introduced the Random Forest algorithm. Citing this shows you're referencing the foundational theory behind your chosen model.
  - On Predictive Maintenance using Machine Learning:
    - Carvalho, T. P., Soares, F. A., Vita, R., Francisco, R. d. P., Basto, J. P., & Alcalá, S. G. (2019). A systematic literature review of machine learning methods applied to predictive maintenance. Computers & Industrial Engineering, 137, 106024.
    - This academic review provides a comprehensive overview of different machine learning methods used for predictive maintenance. It helps contextualize your work and justifies the use of a classification approach.

edunet foundation

IBM CERTIFICATIONS



Getting Started with Artificial Intelligence

IBM SkillsBuild

IBM

In recognition of the commitment to achieve professional excellence

Swapnil Patidar

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/7212526a-742b-44f5-9ed5-d3f6b7f1c3bf

IBM

IBM CERTIFICATIONS

In recognition of the commitment to achieve professional excellence

Journey to Cloud: Envisioning Your Solution
IBM SkillsBuild

Swapnil Patidar

Has successfully satisfied the requirements for:

Journey to Cloud: Envisioning Your Solution

Issued on: Jul 24, 2025
Issued by: IBM SkillsBuild

Verify: https://www.credly.com/badges/1cdeae1a-734f-4f75-94dd-47425cac9d62

IBM

IBM CERTIFICATIONS

IBM **SkillsBuild** Completion Certificate

This certificate is presented to

Swapnil Patidar

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