CAPSTONE PROJECT

PREDICTIVE MAINTENANCE OF INDUSTRIAL MACHINERY

Presented By:
1. Tarun Jayant V M - Sri Ramakrishna Engineering College - CSE

OUTLINE

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

PROBLEM STATEMENT

Industrial machinery is prone to sudden failures such as tool wear, overheating, and power disruptions, leading to unplanned downtime and costly repairs. Predictive maintenance strategies aim to anticipate these failures by analyzing operational data, allowing maintenance before failure occurs. However, identifying patterns across various failure types remains a challenge due to complex multi-class classification requirements and real-time data variability.

PROPOSED SOLUTION

- **The proposed system aims to address the challenge of predicting potential failures in industrial machines to ensure proactive maintenance and minimize downtime. This involves leveraging machine learning techniques to classify different types of machinery failures based on real-time operational sensor data. The solution will consist of the following components:**

1) **Data Collection:**

- Gather historical sensor data from industrial machines, including parameters like torque, rotational speed, tool wear, and machine failure labels.
- Utilize real-time operational logs from industrial environments to enrich the dataset and simulate realistic failure conditions.

2) **Data Preprocessing:**

- Clean and preprocess the dataset to handle missing values, anomalies, and outliers to improve model performance.
- Use automated feature selection and transformation via IBM Watsonx.ai to identify relevant features contributing to failure prediction.

3) **Machine Learning Algorithm:**

- Implement a multi-class classification model using **Snap Random Forest (Batched Tree Ensemble Classifier)** automatically selected by IBM Watsonx.ai.
- Consider all failure types such as **Tool Wear, Heat Dissipation, Overstrain, Power Failure, Random Failures**, and **No Failure** to ensure complete fault coverage.

4) **Deployment:**

- Use IBM Watsonx.ai pipelines to generate, evaluate, and rank model candidates with minimal manual effort.
- Export the best-performing model pipeline for potential deployment in edge systems, dashboards, or cloud monitoring solutions for real-time failure prediction.

5) **Evaluation:**

- Evaluate model performance using ROC curve, confusion matrix, and precision-recall metrics.
- Achieved 99.7% accuracy using holdout validation in IBM Watsonx.ai

SYSTEM APPROACH

System Requirements:

- **IBM Cloud Account (Lite Plan)**
  Used to access IBM Watsonx.ai Studio for building the model.
- **IBM Watsonx.ai Studio**
  Environment Resource Usage
- **1 vCPU + 4 GB RAM** → 0.5 capacity units
- **2 vCPU + 8 GB RAM** → 1 capacity unit
- **4 vCPU + 16 GB RAM** → 2 capacity units
- A cloud-based visual AI development tool used to create and train the AutoAI pipelines for multi-class classification.
- **CSV Dataset from Kaggle**
  The predictive maintenance dataset containing sensor values and failure types was used as the input data source.

Libraries Required to Build the Model:

- Although AutoAI automates model development without needing to manually import libraries, it internally uses the following:
- **Pandas & NumPy**
  For internal data handling, transformation, and feature extraction.
- **Scikit-learn**
  For model training, validation, and metrics (confusion matrix, ROC, precision-recall).
- **Snap ML (IBM's Accelerated Library)**
  For fast and efficient training of ensemble models like the Batched Tree Ensemble Classifier.
- **Matplotlib / Seaborn (Visuals)**
  Used by Watsonx.ai to generate visual outputs like ROC curves, confusion matrices, and precision-recall plots.

ALGORITHM & DEPLOYMENT

Algorithm Selection:

- IBM Watsonx.ai AutoAI selected the **Batched Tree Ensemble Classifier (Snap Random Forest)**, optimized for high accuracy and fast training on structured, tabular sensor data.
- This algorithm is suitable for **multi-class classification**, effectively distinguishing different types of machine failures.

Data Input:

- Input features include **rotational speed, torque, tool wear**, and **operational settings.**
- The target output is the **type of failure**, such as Tool Wear, Power Failure, or No Failure.

Training Process:

- The Watsonx.ai AutoAI engine automatically performed **data preprocessing**, **feature engineering**, and **model tuning**.
- Multiple models were trained and ranked using **holdout validation**, selecting the pipeline with **99.7% accuracy**.

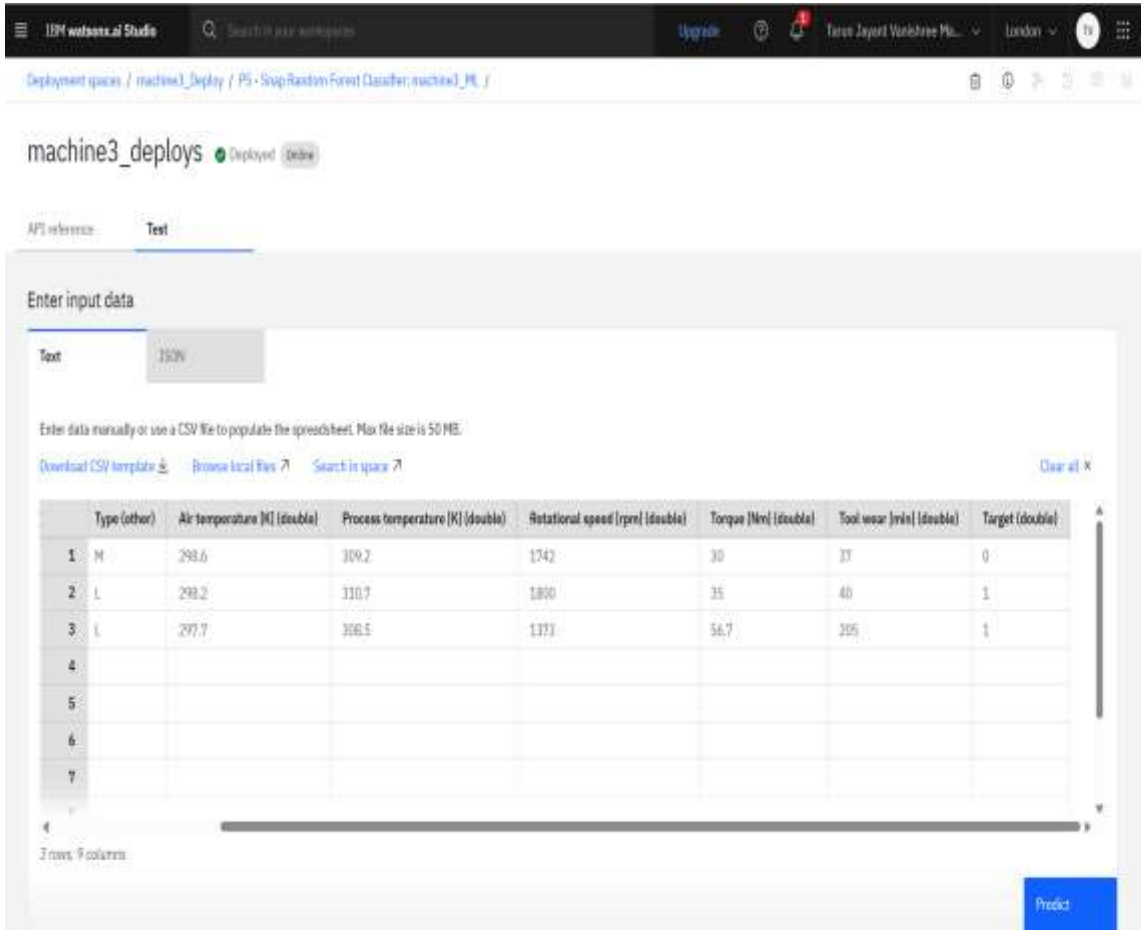
Prediction Process:

- The final model predicts the **specific failure category** based on live sensor inputs.
- It supports **real-time prediction** and can be integrated with dashboards or alerts for maintenance teams.

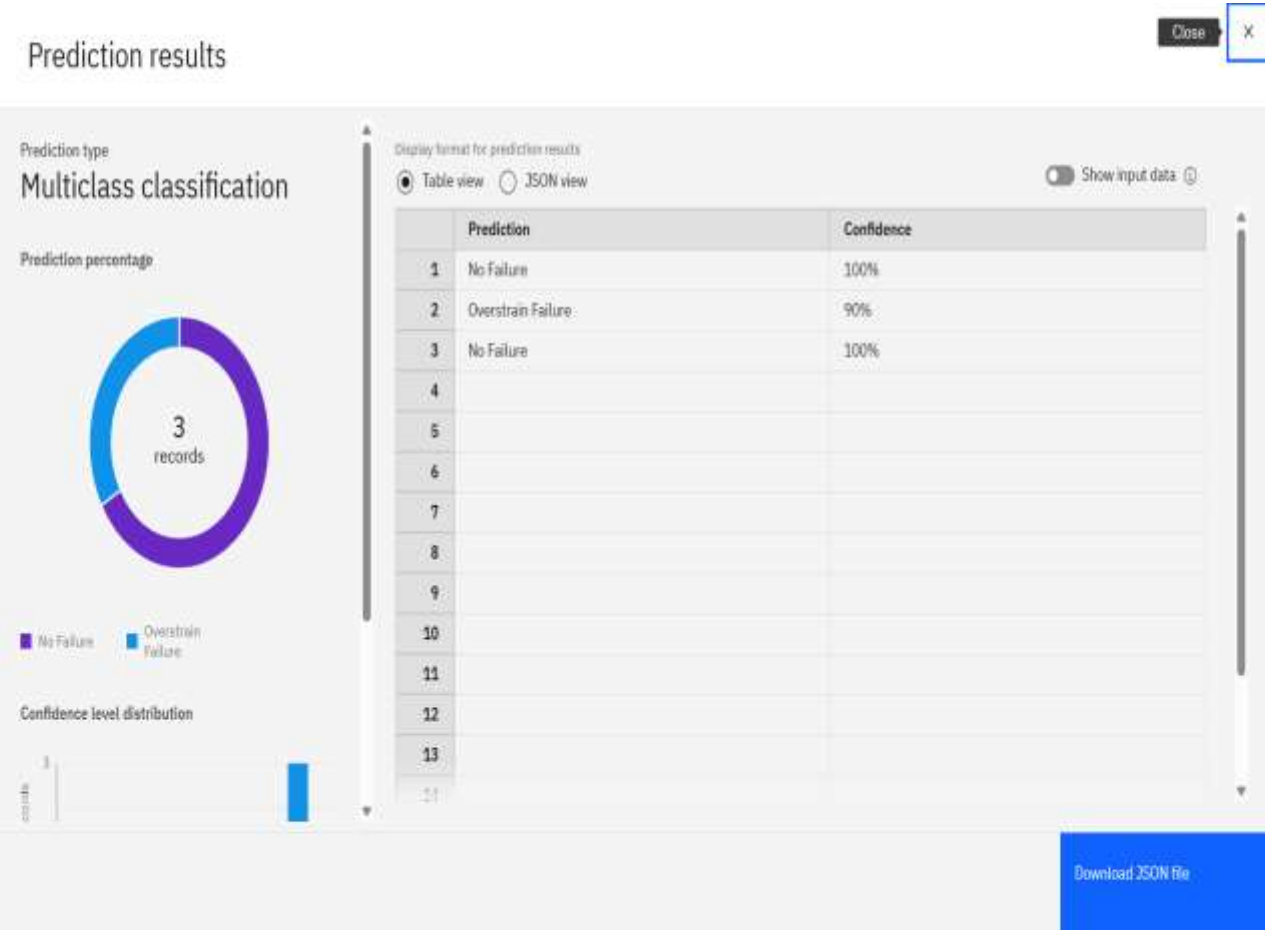
Deployment:

- The trained model pipeline from IBM Watsonx.ai was exported and deployed as a **REST API** via IBM Cloud. It can now be integrated into industrial systems for **real-time failure prediction and alerting.**

RESULT



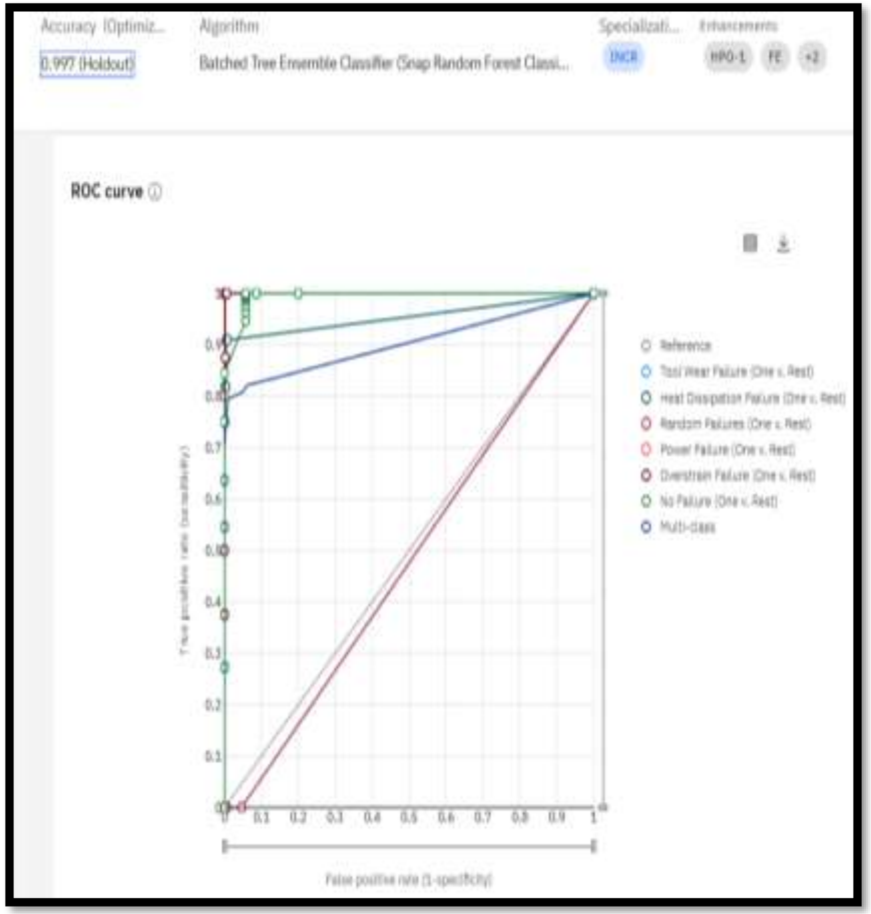
IBM watsonx.ai Studio
Upgrade
London
machine3_deploys
Deployed
Online
API reference
Test
Enter input data
Text
JSON
Enter data manually or use a CSV file to populate the spreadsheet. Max file size is 50 MB.
Download CSV template
Browse local files
Search in space
Clear all
Type (other)
Air temperature [K] (double)
Process temperature [K] (double)
Rotational speed [rpm] (double)
Torque [Nm] (double)
Tool wear [min] (double)
Target (double)
1 M 298.6 309.2 1742 30 37 0
2 L 298.2 310.7 1800 35 40 1
3 L 297.7 308.5 1373 56.7 205 1
3 rows, 9 columns
Predict



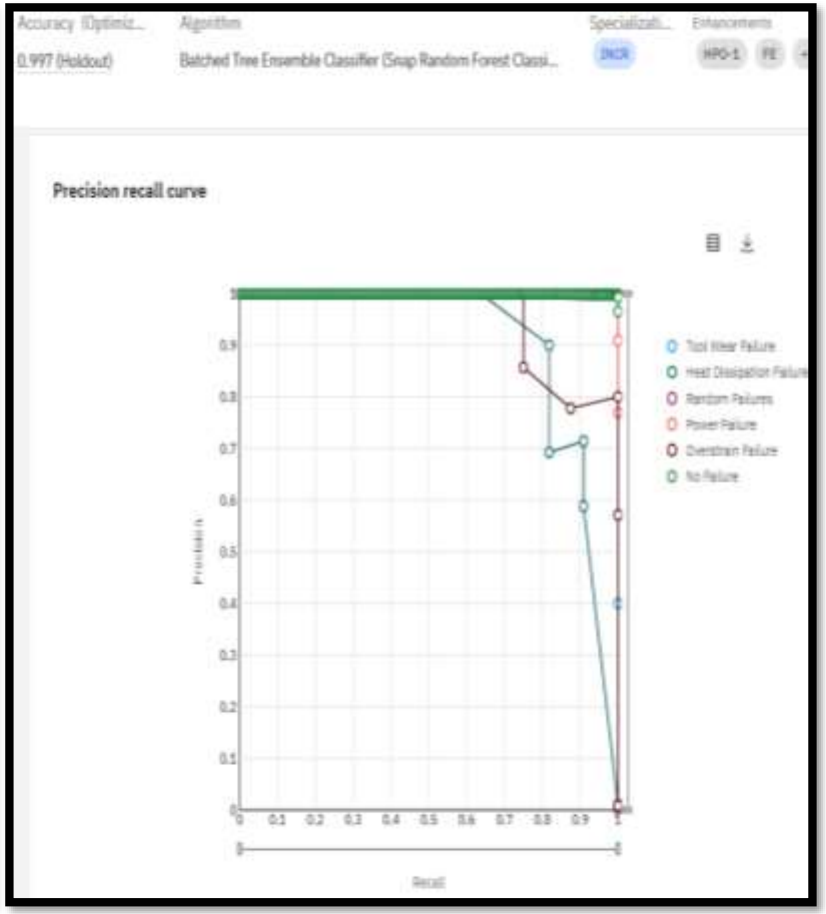
Prediction results
Close
Prediction type
Multiclass classification
Prediction percentage
3
records
No Failure
Overstrain Failure
Confidence level distribution
Table view
JSON view
Show input data
Prediction
Confidence
1 No Failure 100%
2 Overstrain Failure 90%
3 No Failure 100%
Download JSON file

RESULT

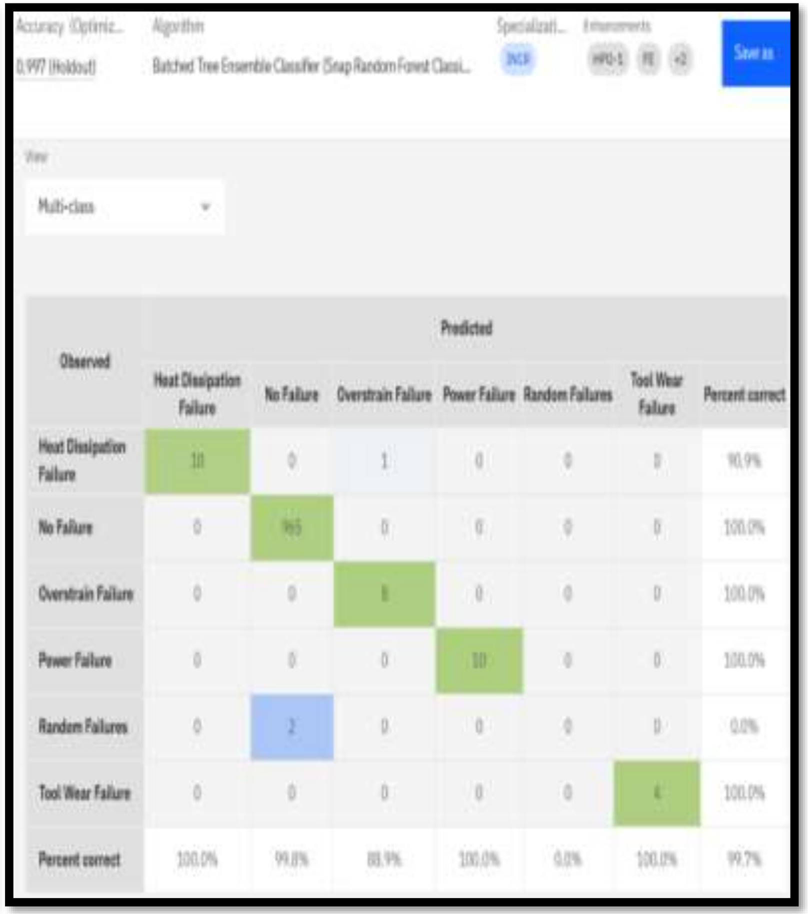
ROC Curve



Precision-Recall



Confusion Matrix



Observed	Heat Dissipation Failure	No Failure	Overstrain Failure	Power Failure	Random Failures	Tool Wear Failure	Percent correct
Heat Dissipation Failure	10	0	1	0	0	0	90.9%
No Failure	0	965	0	0	0	0	100.0%
Overstrain Failure	0	0	8	0	0	0	100.0%
Power Failure	0	0	0	10	0	0	100.0%
Random Failures	0	2	0	0	0	0	0.0%
Tool Wear Failure	0	0	0	0	0	4	100.0%
Percent correct	100.0%	99.8%	88.9%	100.0%	0.0%	100.0%	99.7%

CONCLUSION

- The predictive maintenance model built using **IBM Watsonx.ai** achieved a high **accuracy of 99.7%**, effectively classifying various machinery failure types. AutoAI streamlined the process from data preprocessing to model selection, enabling quick and efficient development.
- While the model performed well overall, handling **imbalanced classes** like random failures posed a minor challenge. Future improvements include real-time data integration and model explainability.
- Accurate failure prediction plays a vital role in industrial operations by minimizing unexpected breakdowns, reducing downtime, and lowering maintenance costs—ensuring **operational continuity and safety** in machinery-dependent environments.

FUTURE SCOPE

- **Incorporate Real-Time Sensor Streams:** Integrate live IoT data from industrial machines for continuous monitoring and faster failure prediction.
- **Enhance Data Diversity:** Include additional contextual data like environmental conditions, usage frequency, or historical repair logs to improve model robustness.
- **Optimize Model Performance:** Fine-tune the algorithm using advanced techniques like hyperparameter tuning, ensemble stacking, or feature importance analysis for even higher accuracy.
- **Scale Across Industries:** Extend the system to different machinery types and industrial sectors, or deploy it across **multiple facilities and regions**.
- **Adopt Emerging Technologies:** Explore **edge computing** for on-device predictions and use **deep learning models** to improve model transparency and decision-making.

edunet foundation

REFERENCES

- **Scikit-learn Developers**, *Model Evaluation: Precision, Recall, F-score, and Support*, Scikit-learn Documentation.

https://scikit-learn.org/stable/modules/model_evaluation.html

- **Snap ML – IBM Research**, *Accelerated Machine Learning Library for Cloud and Edge*.

https://snapml.readthedocs.io

- **Shivam Bansal**, *Machine Predictive Maintenance Dataset*, Kaggle.

https://www.kaggle.com/datasets/shivamb/machine-predictive-maintenance-classification

- **IBM Documentation**, *AutoAI in Watsonx.ai Studio*, IBM Cloud Docs.

https://www.ibm.com/docs/en/cloud-paks/cp-data/4.6.x?topic=tools-autoai

IBM CERTIFICATIONS

In recognition of the commitment to achieve professional excellence

Getting Started with Artificial Intelligence

IBM SkillsBuild

Tarun Jayant V M

Has successfully satisfied the requirements for:

Getting Started with Artificial Intelligence

Issued on: Jul 20, 2025
Issued by: IBM SkillsBuild

Verify: https://www.credly.com/badges/6b7c65a7-1c3b-4e39-87ef-28ac8d761cf9

IBM

IBM CERTIFICATIONS


In recognition of the commitment to achieve professional excellence

Journey to Cloud: Envisioning Your Solution

IBM

Tarun Jayant V M

Has successfully satisfied the requirements for:

Journey to Cloud: Envisioning Your Solution

Issued on: Jul 20, 2025

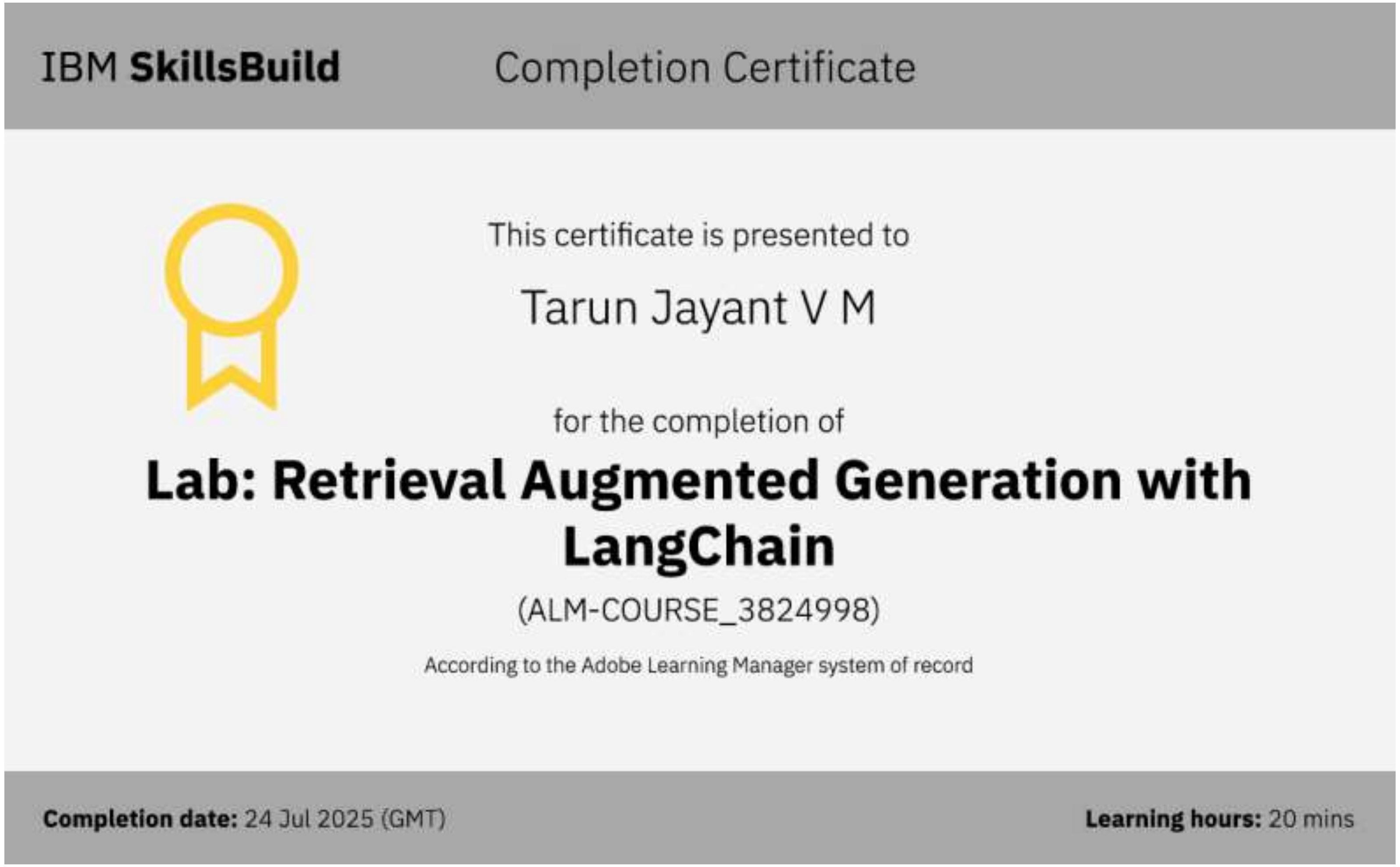
Issued by: IBM SkillsBuild

Verify: https://www.credly.com/badges/667eb2a7-0b8e-45a6-8e59-3019869dbc7b

IBM

edunet foundation

IBM CERTIFICATIONS


IBM **SkillsBuild** Completion Certificate

This certificate is presented to

Tarun Jayant V M

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