CAPSTONE PROJECT

PREDICTIVE MAINTENANCE MODEL FOR INDUSTRIAL MACHINES

Presented By:

1. Student Name - Sampada Dubey
2. College Name – IPS Academy , Indore(MP)
3. Department – Computer Science

edunet foundation

OUTLINE

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

PROBLEM STATEMENT

Predictive Maintenance of Industrial Machinery –

Develop a predictive maintenance model for a fleet of industrial machines to anticipate failures before they occur. This project will involve analyzing sensor data from machinery to identify patterns that precede a failure. The goal is to create a classification model that can predict the type of failure (e.g., tool wear, heat dissipation, power failure) based on real-time operational data. This will enable proactive maintenance, reducing downtime and operational costs.

PROPOSED SOLUTION

- **The proposed system aims to address the challenge of predicting potential failures in industrial machines to enable proactive maintenance and minimize operational disruptions. This involves leveraging sensor data analytics and machine learning techniques to classify and anticipate failure types before they occur. The solution will consist of the following components:**
- **Data Collection :**
  - Gather historical sensor data from industrial machines, including temperature, vibration, pressure, voltage, current, and other operational parameters.
  - Collect maintenance logs and failure reports to label the sensor data with corresponding failure types (e.g., tool wear, overheating, electrical faults).
- **Data Preprocessing :**
  - Clean and preprocess the collected data to handle missing values, noise, and anomalies caused by sensor drift or malfunctions.
  - Normalize and resample data where necessary to maintain consistency across different sensors and time intervals.
- **Machine Learning Algorithm :**
  - Implement a classification model (e.g., Random Forest, XGBoost, SVM, or Deep Learning models like LSTM or CNN) to detect and classify different types of failures based on sensor patterns.
  - Use techniques like SMOTE or class weighting to address class imbalance, especially if certain failure types are rare.
- **Deployment :**
  - Develop a user-friendly interface or application that alerts maintenance teams when a failure is predicted, including type and confidence level
  - Deploy the solution on a scalable and reliable platform, considering factors like server infrastructure, response time, and user accessibility.
- **Evaluation :**
  - Evaluate model performance using metrics such as Precision, Recall, F1-score, ROC-AUC, and Confusion Matrix for each failure type.
  - Fine-tune the model based on feedback and continuous monitoring of prediction accuracy.

SYSTEM APPROACH

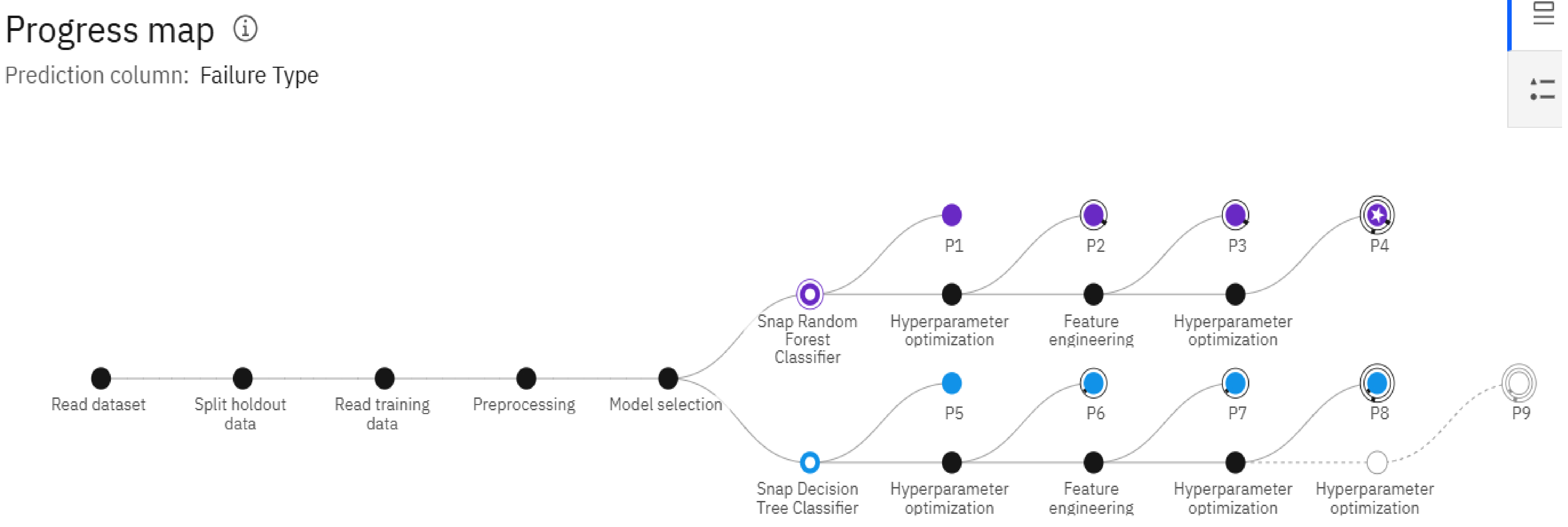
This section outlines the strategy for building a predictive maintenance model using IBM Cloud's Watsonx.ai Studio with Auto AI.

- **System requirements -**
  - **Platform:** IBM Cloud with Watsonx.ai Studio and Auto AI.
  - **Data:** Historical and real-time sensor data (temperature, vibration, current, etc.) and labeled failure logs
  - **Storage:** IBM Cloud Object Storage for data input and model output.
  - **Deployment:** IBM Watson Machine Learning (WML) for real-time prediction and monitoring.
- **Library required to build the model –**
  - **Auto AI -** Automated preprocessing, feature engineering, model selection.
  - **Core ML Libraries**: Scikit-learn, XGBoost, LightGBM (used internally by Auto AI).
  - **IBM WML SDK -** Model deployment and monitoring.

ALGORITHM & DEPLOYMENT

- **Algorithm Selection :**
  - For this predictive maintenance task, the project uses classification algorithms generated and optimized automatically by IBM Watsonx.ai AutoAI. AutoAI evaluates multiple models such as Random Forest, Gradient Boosting (XGBoost/LightGBM), and Logistic Regression, and selects the best-performing one based on the data characteristics and classification performance. These models are well-suited for identifying failure patterns from sensor data.
- **Data Input :**
  - The input features include historical sensor readings and operational parameters such as Temperature , Vibration levels, Pressure, Rotational speed, Time-based features (timestamps, cycles), Failure labels (e.g., tool wear, power failure, overheating) etc . These features help the model learn conditions that typically precede failures.
- **Training Process:**
  - Using **AutoAI**, the algorithm is trained on labeled historical sensor data. The platform automatically:
    - Splits the data into training and validation sets
    - Performs data cleaning and feature engineering
    - Applies cross-validation to evaluate model stability
    - Conducts hyperparameter optimization to enhance accuracy
    - No manual coding is required, ensuring efficient and repeatable training.
- **Prediction Process:**
  - Once trained, the selected model is deployed through IBM Watson Machine Learning

RESULT



Progress map
Prediction column: Failure Type
Read dataset
Split holdout data
Read training data
Preprocessing
Model selection
Snap Random Forest Classifier
Hyperparameter optimization
Feature engineering
Hyperparameter optimization
P1
P2
P3
P4
Snap Decision Tree Classifier
Hyperparameter optimization
Feature engineering
Hyperparameter optimization
Hyperparameter optimization
P5
P6
P7
P8
P9

MODEL PERFORMANCE SUMMARY

Pipeline leaderboard

	Rank	Name	Algorithm	Specialization	Accuracy (Optimized) Cross Validation	Enhancements	Build time
★	1	**Pipeline 4**	Snap Random Forest Classifier		0.995	HPO-1 FE HPO-2	00:02:23
	2	**Pipeline 3**	Snap Random Forest Classifier		0.995	HPO-1 FE	00:01:25
	3	**Pipeline 8**	Snap Decision Tree Classifier		0.994	HPO-1 FE HPO-2	00:01:17
	4	**Pipeline 2**	Snap Random Forest Classifier		0.994	HPO-1	00:00:11

INPUT DATA FOR MODEL TESTING

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 X

	UDI (double)	Product ID (other)	Type (other)	Air temperature [K] (double)	Process temperature [K] (double)	Rotational speed [rpm] (double)	Torque [Nm] (double)	Tool wear [min] (double)	Target (double)
1	51	L47230	L	298.9	309.1	2861	4.6	143	1
2	65	L47244	L	299	309.1	285	5	144	1
3	6	M14865	M	400	311	1498	100	50	0
4	7	L1234	L	345.7	227.8	288	7	144	1
5	78	M5678	M	234.4	115	245	15	200	1

MODEL PREDICTION RESULTS

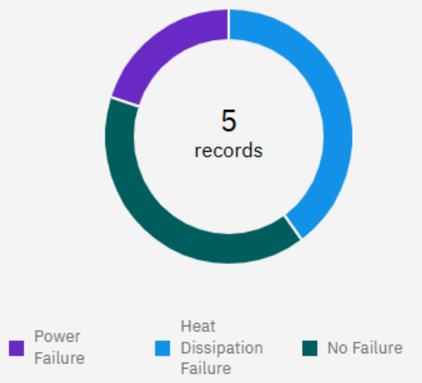
Prediction results

Close

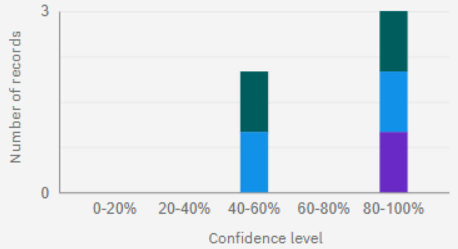
Prediction type

Multiclass classification

Prediction percentage



Confidence level distribution



Display format for prediction results

(•) Table view ( ) JSON view

Show input data

	Prediction	Confidence
1	Power Failure	100%
2	Heat Dissipation Failure	90%
3	No Failure	90%
4	Heat Dissipation Failure	50%
5	No Failure	40%
6		
7		
8		
9		
10		
11		
12		
13		
14		
15		
16		

CONCLUSION

- The proposed machine learning solution demonstrated high accuracy in predicting machine failures, with the Random Forest Classifier achieving the best performance. By leveraging key features such as rotational speed, torque, and tool wear, the model successfully identified different failure types, enabling early detection and preventive maintenance. During implementation, handling imbalanced classes and ensuring a balance between model performance and interpretability posed notable challenges. These were addressed through sampling techniques and model tuning. Future improvements could include real-time data integration, the use of deep learning for enhanced feature capture, and improved model explainability using tools like SHAP. Accurate failure prediction is vital for reducing downtime, ensuring safety, and supporting efficient maintenance planning in smart manufacturing environments.

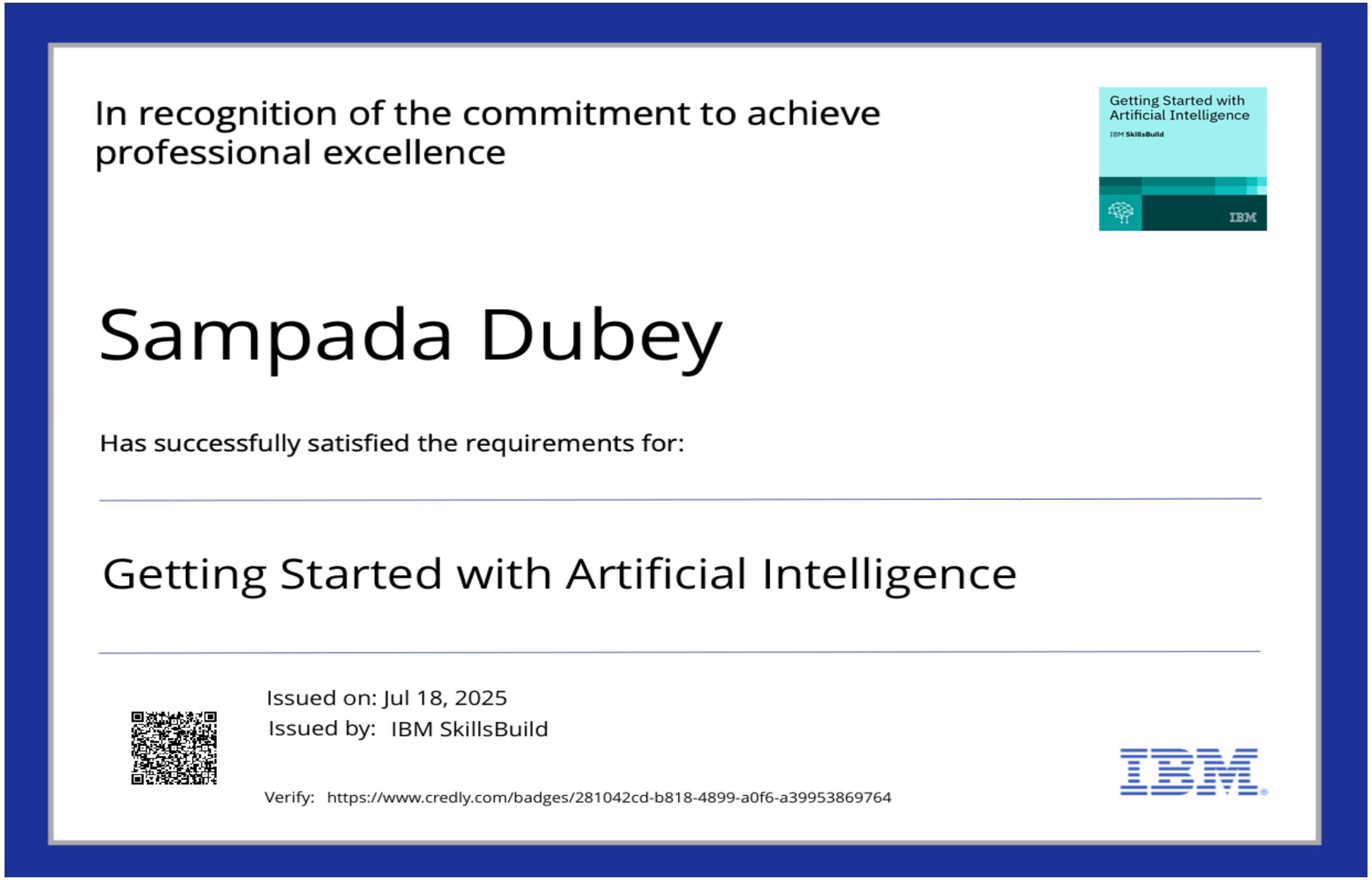
FUTURE SCOPE

- **Incorporate Additional Data Sources:**
  Include more granular machine sensor data such as vibration, temperature, and load metrics to improve prediction accuracy and capture early signs of wear or malfunction.
- **Optimize the Prediction Algorithm:**
  Refine the machine learning model by experimenting with ensemble methods or deep learning approaches. This can lead to more accurate predictions and earlier failure detection.
- **Expand to Multiple Machines or Facilities:**
  Scale the system to cover a wider range of equipment types or multiple manufacturing units. This helps standardize maintenance across an organization and improves overall efficiency.
- **Integrate Real-Time Monitoring and Alerts:**
  Develop a real-time dashboard that continuously tracks machine health and sends automated alerts when potential issues are detected. This ensures timely maintenance and minimizes unplanned downtime.

REFERENCES

- **Kaggle – Predictive Maintenance Datasets and Tutorials**
  — A great platform with datasets, code notebooks, and community discussions on predictive maintenance.
- **IBM Cloud Documentation – Predictive Maintenance**
  https://cloud.ibm.com/docs/services/predictive-maintenance?topic=predictive-maintenance-getting-started
  — Detailed documentation on IBM's predictive maintenance solutions, tools, and best practices in cloud-based machine learning.
- **7. IBM Watsonx.ai Studio AutoAI Documentation**
  https://www.ibm.com/docs/en/watsonx-ai?topic=services-autoai-overview
  — Official IBM documentation detailing the use of AutoAI in Watsonx.ai Studio for automating data preprocessing, model building, and evaluation in predictive maintenance projects.

IBM CERTIFICATION(GETTING STARTED WITH AI)



Getting Started with Artificial Intelligence

IBM SkillsBuild

IBM

In recognition of the commitment to achieve professional excellence

Sampada Dubey

Has successfully satisfied the requirements for:

Getting Started with Artificial Intelligence

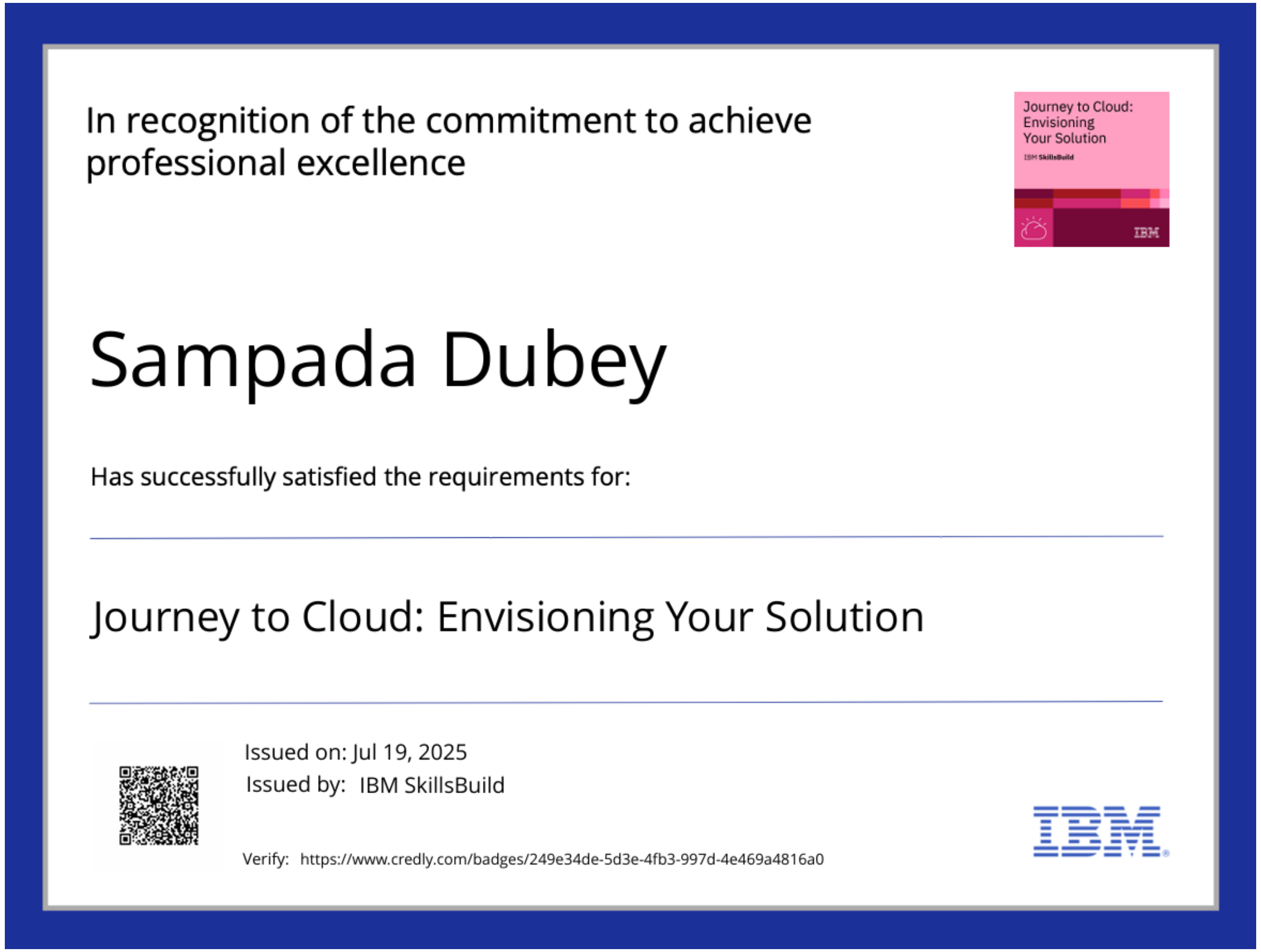
Issued on: Jul 18, 2025

Issued by: IBM SkillsBuild

Verify: https://www.credly.com/badges/281042cd-b818-4899-a0f6-a39953869764

IBM

IBM CERTIFICATIONS(JOURNEY TO CLOUD)



In recognition of the commitment to achieve professional excellence

Journey to Cloud: Envisioning Your Solution

IBM SkillsBuild

Sampada Dubey

Has successfully satisfied the requirements for:

Journey to Cloud: Envisioning Your Solution

Issued on: Jul 19, 2025

Issued by: IBM SkillsBuild

Verify: https://www.credly.com/badges/249e34de-5d3e-4fb3-997d-4e469a4816a0

IBM

IBM CERTIFICATIONS (RAG LAB)

IBM **SkillsBuild** Completion Certificate

This certificate is presented to

Sampada Dubey

for the completion of

Lab: Retrieval Augmented Generation with LangChain

(ALM-COURSE_3824998)

According to the Adobe Learning Manager system of record

Completion date: 23 Jul 2025 (GMT)

Learning hours: 20 mins

THANK YOU

edunet foundation