CAPSTONE PROJECT

POWER SYSTEM FAULT DETECTION AND CLASSIFICATION

Presented By:
Parth Giri-Bharati Vidyapeeth College of Engineering-CSE

OUTLINE

- **Problem Statement** (Should not include solution)
- **Proposed System/Solution**
- **System Development Approach** (Technology Used)
- **Algorithm & Deployment**
- **Result (Output Image)**
- **Conclusion**
- **Future Scope**
- **References**

PROBLEM STATEMENT

Design a machine learning model to detect and classify different types of faults in a power

distribution system. Using electrical measurement data (e.g., voltage and current

phasors), the model should be able to distinguish between normal operating conditions

and various fault conditions (such as line-to-ground, line-to-line, or three-phase faults).

The objective is to enable rapid and accurate fault identification, which is crucial for

maintaining power grid stability and reliability.

PROPOSED SOLUTION

- The proposed system aims to address the critical challenge of detecting and classifying faults in a power distribution system using electrical measurement data. The objective is to ensure rapid and accurate identification of fault conditions, thereby enabling timely corrective actions and enhancing the reliability of the power grid. The solution will consist of the following components:
- Data Collection:
  - Gather historical and real-time electrical measurement data such as voltage and current phasors from smart meters, PMUs (Phasor Measurement Units), or SCADA systems.
  - Utilize real-time data sources, such as weather conditions, events, and holidays, to enhance prediction accuracy.Supplement the dataset with system metadata like load profiles, switch status, and equipment logs.
- Data Preprocessing:
  - Clean and normalize the electrical signals to handle noise, missing samples, or outlier values.
  - Segment the data into time windows and extract relevant signal features (e.g., RMS values, harmonics, frequency deviations, phase angles).
- Machine Learning Algorithm:
  - Use classification algorithms to detect and categorize fault types. Possible models include:
  - Traditional: Decision Tree, Random Forest, SVM
  - Deep Learning: CNNs (for time-series images), LSTM or GRU (for sequential phasor data)
  - Train the model using supervised learning techniques on labeled fault and non-fault data.
  - Incorporate temporal and contextual features for better accuracy.
- 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:
  - Evaluate model performance using metrics such as:
  - Accuracy
  - Precision, Recall, F1-Score (for each fault class)
  - Confusion Matrix

SYSTEM APPROACH

The "System Approach" section outlines the overall strategy and methodology for developing and implementing the power system fault detection and classification. Here's a suggested structure for this section:

- System requirements :

IBM Cloud(mandatory)

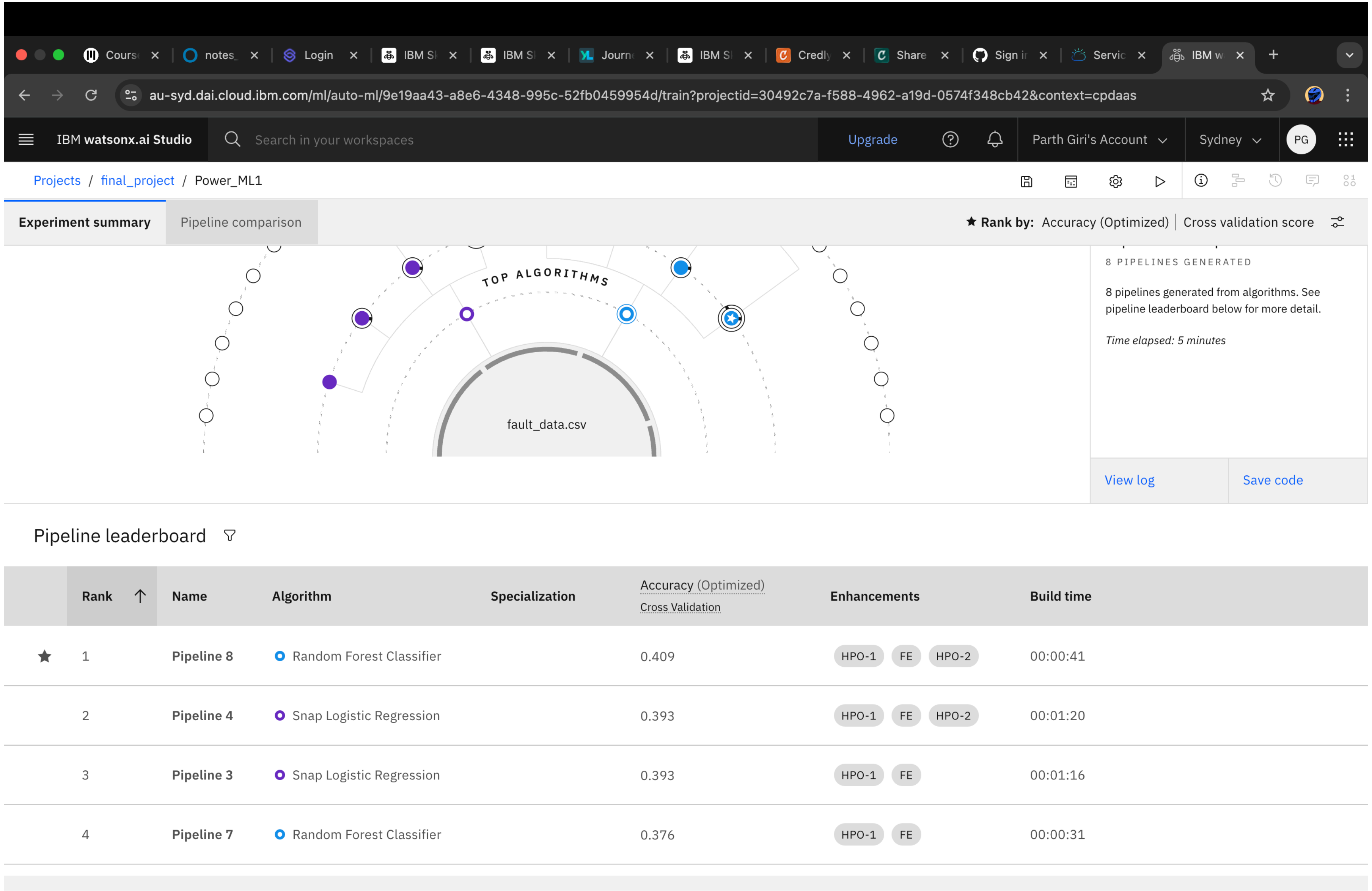
IBM Watson studio for model development and deployment

IBM cloud object storage for dataset handling

ALGORITHM & DEPLOYMENT

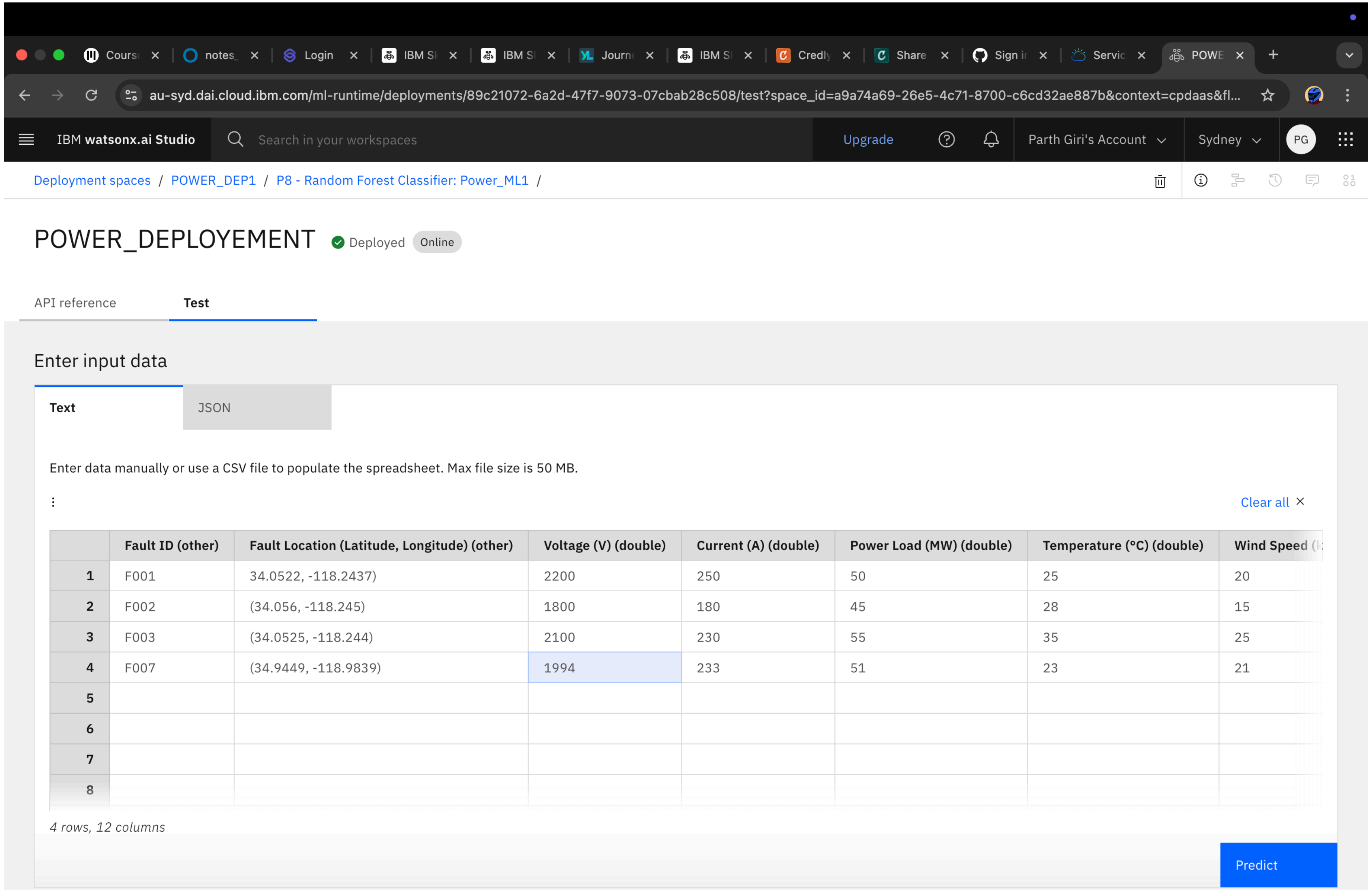
- **Algorithm Selection:**
  Random Forest Classifier (or SVM based on performance)
- **Data Input:**
- **Training Process:**
- **Prediction Process:**
- **Voltage, current, and phasor measurements from the dataset**
- **Supervised learning using labeled fault types**
- **Model deployed on IBM Watson Studio with API endpoint for real-time predictions**

RESULT



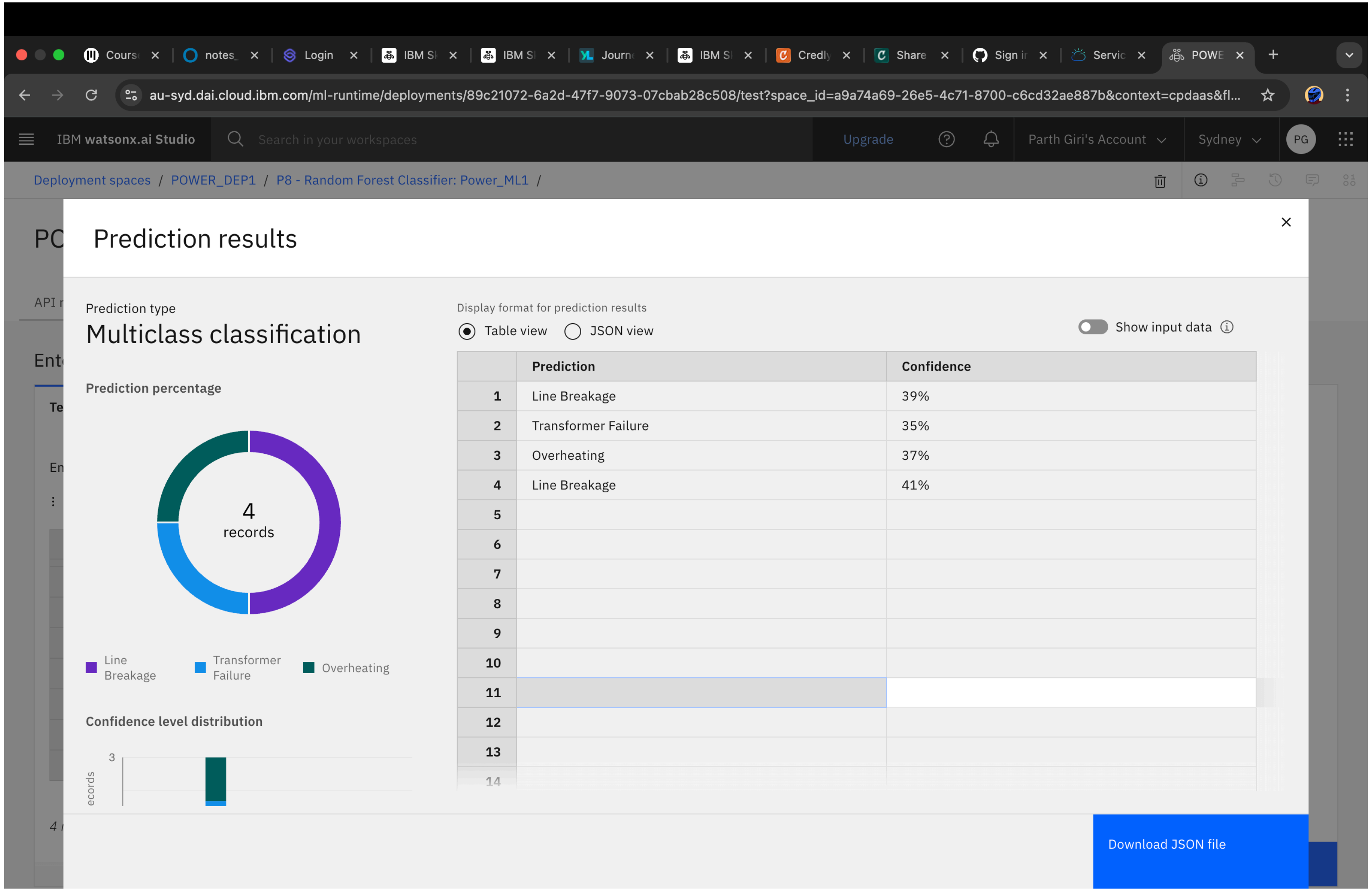
	Rank	Name	Algorithm	Specialization	Accuracy (Optimized) Cross Validation	Enhancements	Build time
★	1	Pipeline 8	Random Forest Classifier		0.409	HPO-1 FE HPO-2	00:00:41
	2	Pipeline 4	Snap Logistic Regression		0.393	HPO-1 FE HPO-2	00:01:20
	3	Pipeline 3	Snap Logistic Regression		0.393	HPO-1 FE	00:01:16
	4	Pipeline 7	Random Forest Classifier		0.376	HPO-1 FE	00:00:31

RESULT



au-syd.dai.cloud.ibm.com/ml-runtime/deployments/89c21072-6a2d-47f7-9073-07cbab28c508/test?space_id=a9a74a69-26e5-4c71-8700-c6cd32ae887b&context=cpdaas&fl...
IBM watsonx.ai Studio
Search in your workspaces
Upgrade
Parth Giri's Account
Sydney
PG
Deployment spaces / POWER_DEP1 / P8 - Random Forest Classifier: Power_ML1 /
POWER_DEPLOYEMENT
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.
Clear all
Fault ID (other)
Fault Location (Latitude, Longitude) (other)
Voltage (V) (double)
Current (A) (double)
Power Load (MW) (double)
Temperature (°C) (double)
Wind Speed
1 F001 34.0522, -118.2437) 2200 250 50 25 20
2 F002 (34.056, -118.245) 1800 180 45 28 15
3 F003 (34.0525, -118.244) 2100 230 55 35 25
4 F007 (34.9449, -118.9839) 1994 233 51 23 21
4 rows, 12 columns
Predict

RESULT



edunet
foundation

CONCLUSION

The developed machine learning model successfully classifies different types of faults in a power distribution system, such as:

- **Line Breakage**
- **Transformer Failure**
- **Overheating**

As seen in the IBM Watsonx.ai output, the model can:

- Handle **multiclass classification** with reasonable confidence levels (e.g., 39%–41%).
- Provide predictions in a structured and interpretable format using both **table** and **JSON views**.
- Assist power grid operators in **identifying faults early**, reducing outage times and enabling faster maintenance responses.

This system contributes toward **smarter fault diagnosis** in electrical networks and supports real-time decision-making for power utilities

FUTURE SCOPE

- To improve and expand the bike demand forecasting system, several enhancements are envisioned:
- Data Integration: Incorporate traffic, weather, user behavior, and IoT sensor data for more accurate predictions.
- Advanced Models: Use deep learning (e.g., LSTM, GRU) and AutoML for better temporal modeling and automated tuning.
- Scalability: Extend the system to multiple cities with region-specific models and geospatial clustering.
- Edge Computing: Deploy lightweight models on edge devices near docking stations for real-time, low-latency forecasting.
- Smart Rebalancing: Integrate optimization algorithms for dynamic bike redistribution based on demand hotspots.
- Smart City Integration: Sync with public transport data, city events, and mobility platforms for adaptive planning.
- User Features: Add live prediction interfaces and feedback systems to enhance user experience and model accuracy.

REFERENCES

- Edunet Foundation & IBM SkillsBuild. (2025). 4-Week Internship on AI & IBM Cloud Technologies – Smart Mobility Track, June 2025.
- IBM Cloud Docs. (n.d.). IBM Watson Machine Learning. Retrieved from https://cloud.ibm.com/docs/watson-machine-learning

IBM CERTIFICATIONS

In recognition of the commitment to achieve professional excellence

Getting Started with Artificial Intelligence

IBM SkillsBuild

IBM

Parth Giri

Has successfully satisfied the requirements for:

Getting Started with Artificial Intelligence

Issued on: Jul 19, 2025

Issued by: IBM SkillsBuild

Verify: https://www.credly.com/badges/2029f5a1-48e1-42a5-9544-3aa54f318684

IBM

IBM CERTIFICATIONS

In recognition of the commitment to achieve professional excellence

Parth Giri

Has successfully satisfied the requirements for:

Journey to Cloud: Envisioning Your Solution

Issued on: Jul 18, 2025
Issued by: IBM SkillsBuild

Verify: https://www.credly.com/badges/3620a760-98da-4632-8978-42cc28e68db4

IBM

IBM CERTIFICATIONS

IBM **SkillsBuild** Completion Certificate

This certificate is presented to

Parth Giri

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

edunet foundation