CAPSTONE PROJECT

POWER SYSTEM FAULT DETECTION AND CLASSIFICATION

Presented By:
1. Ashutosh Kumar Soni-Shri Shankaracharya Group of Institutions-CSE

edunet foundation

OUTLINE

- Problem Statement
- Proposed System/Solution
- System Development Approach
- 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 automate the process of detecting and classifying power system faults to ensure timely and reliable operation of the grid. It leverages electrical phasor data (voltage and current) and machine learning techniques to accurately identify various fault types. The solution consists of the following components:
- Data Collection:
  - Use the publicly available Power System Faults Dataset from Kaggle, which contains labeled examples of different fault types and normal operating conditions.
  - The dataset includes voltage and current phasor measurements essential for fault classification(e.g., line-to-ground, line-to-line, three-phase faults).
- Data Preprocessing:
  - Clean and normalize the data to handle missing values, noise, and inconsistencies.
  - Extract relevant features (e.g., phase differences, magnitude deviations) that indicate fault conditions.
- Machine Learning Algorithm:
  - Implement a classification algorithm such as Random Forest, SVM, or Neural Networks to distinguish between different fault types.
  - Train the model using labeled fault data and validate its accuracy using cross-validation techniques.
- Deployment:
  - Develop an interactive dashboard or tool for real-time fault monitoring and classification.
  - Deploy the model on IBM Cloud Lite, ensuring scalability, low latency, and secure access.
- Evaluation:
  - Evaluate the model using metrics like Accuracy, Precision, Recall, and Confusion Matrix.
  - Continuously fine-tune the model with feedback and real-time data for improved fault prediction performance.

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 Watsonx.ai Studio for model development and deployment.

  IBM Cloud Object Storage for dataset handling.

  AutoAI - Automate pipeline creation and optimization.

- User Environment

  Web browser (Chrome/Firefox recommended), stable internet connection

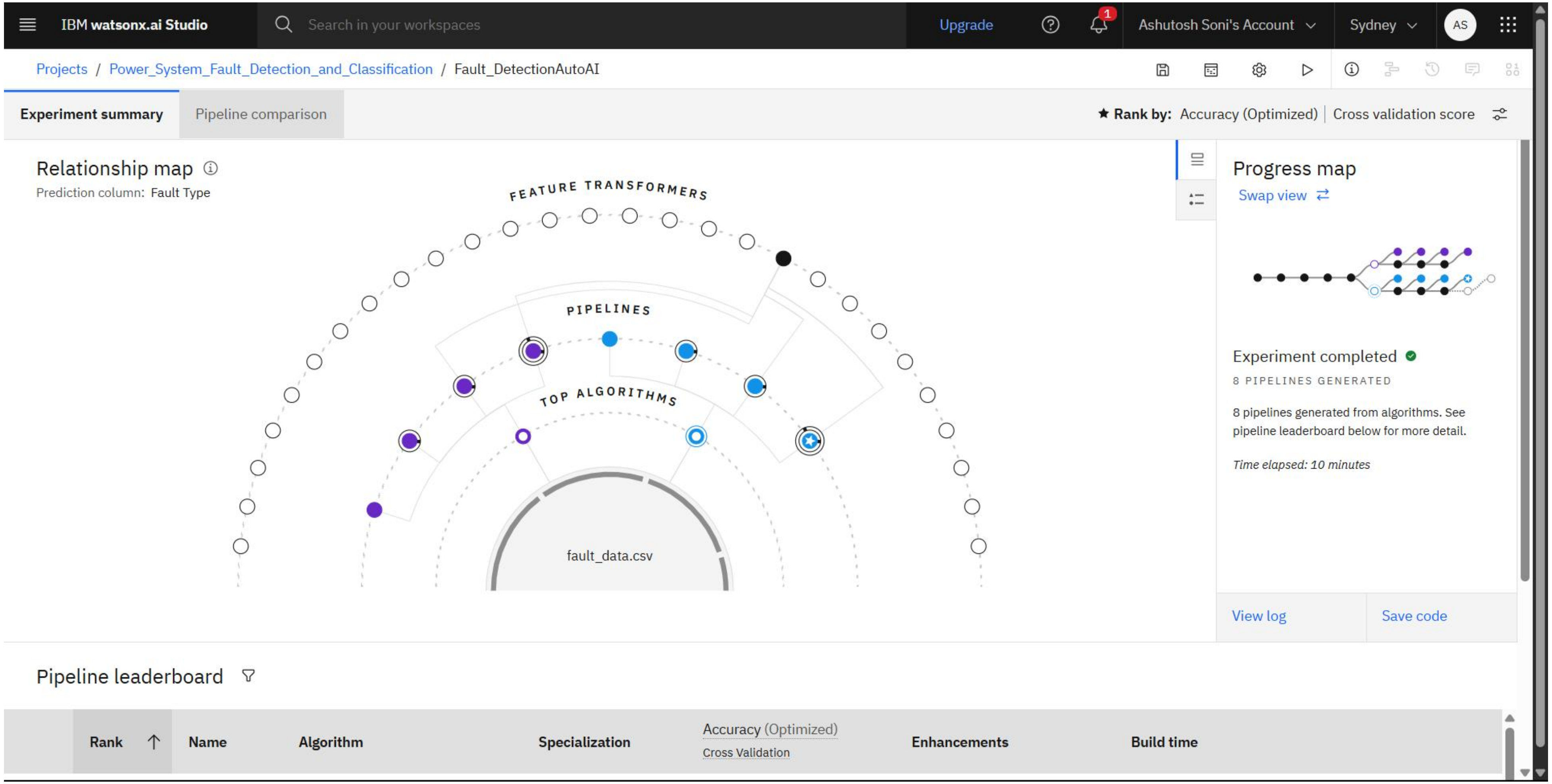
  RAM: 8 GB (minimum)

  OS: Windows/macOS/Linux

ALGORITHM & DEPLOYMENT

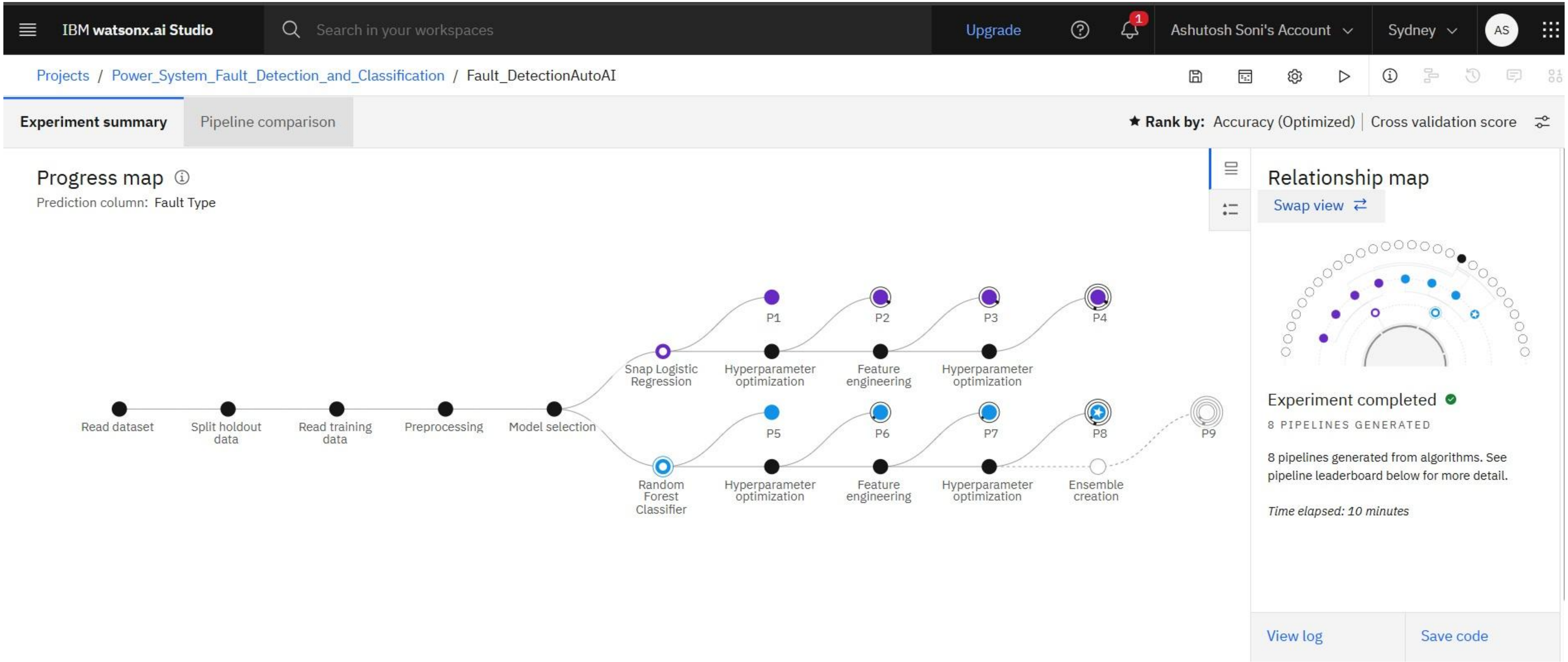
- Algorithm Selection:
  - Used Random Forest and SVM for classification.
  - Chosen for high accuracy and noise tolerance.
- Data Input:
  - Voltage & current features (magnitude, angle).
  - Labeled data with different fault types.
- Training Process:
  - Dataset split into training and testing sets
  - GUI-based training with cross-validation
- Prediction Process:
- Model deployed on Watsonx.ai Studio with API endpoint for real-time predictions.
- The GUI allows easy upload and prediction using the built model.

RESULT



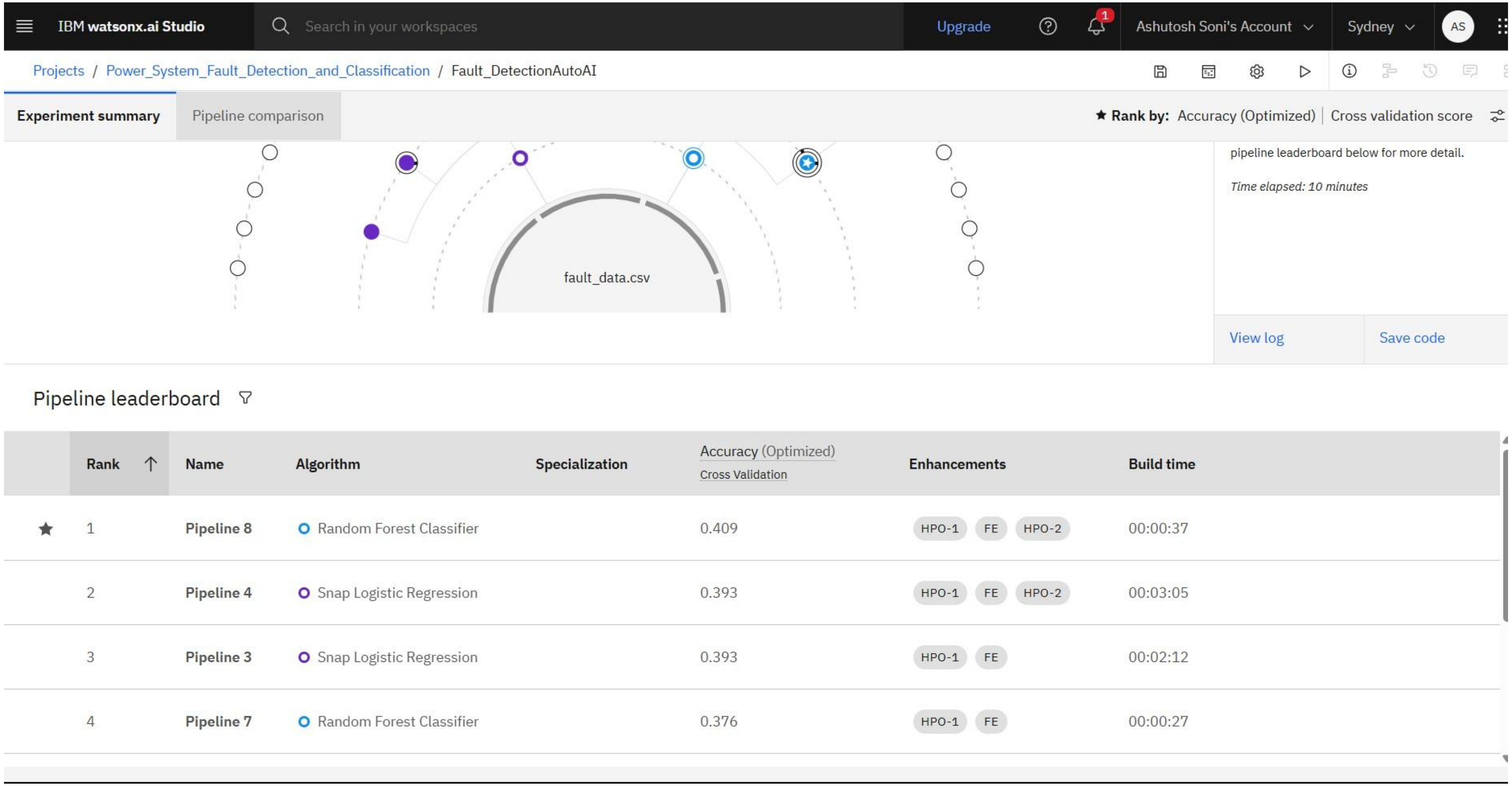
IBM watsonx.ai Studio
Search in your workspaces
Upgrade
Ashutosh Soni's Account
Sydney
AS
Projects / Power_System_Fault_Detection_and_Classification / Fault_DetectionAutoAI
Experiment summary
Pipeline comparison
Rank by: Accuracy (Optimized) | Cross validation score
Relationship map
Prediction column: Fault Type
FEATURE TRANSFORMERS
PIPELINES
TOP ALGORITHMS
fault_data.csv
Progress map
Swap view
Experiment completed
8 PIPELINES GENERATED
8 pipelines generated from algorithms. See pipeline leaderboard below for more detail.
Time elapsed: 10 minutes
View log
Save code
Pipeline leaderboard
Rank
Name
Algorithm
Specialization
Accuracy (Optimized)
Cross Validation
Enhancements
Build time

RESULT



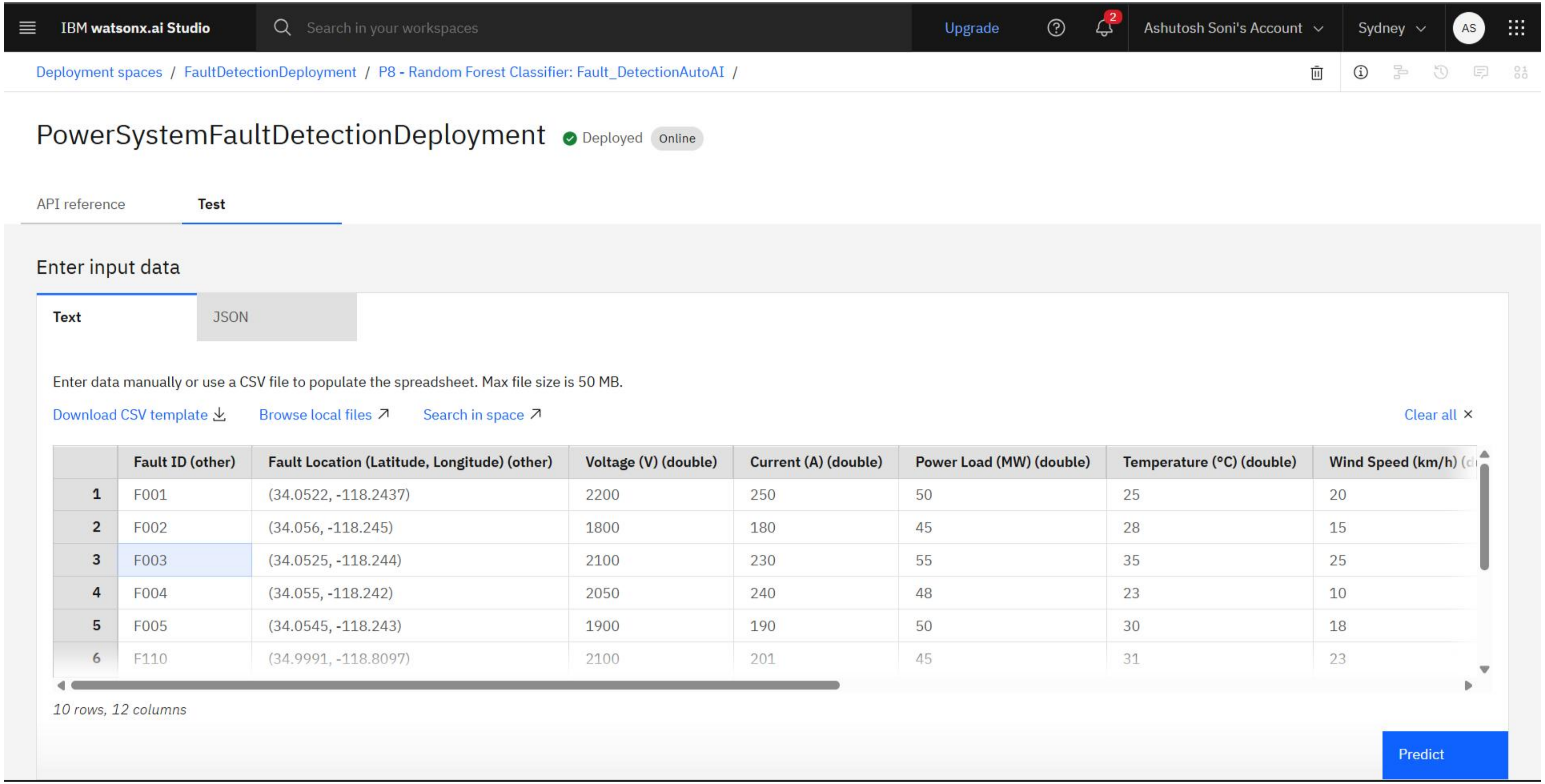
edunet foundation

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	00:00:37
	2	Pipeline 4	Snap Logistic Regression		0.393	HPO-1 FE HPO-2	00:03:05
	3	Pipeline 3	Snap Logistic Regression		0.393	HPO-1 FE	00:02:12
	4	Pipeline 7	Random Forest Classifier		0.376	HPO-1 FE	00:00:27

RESULT

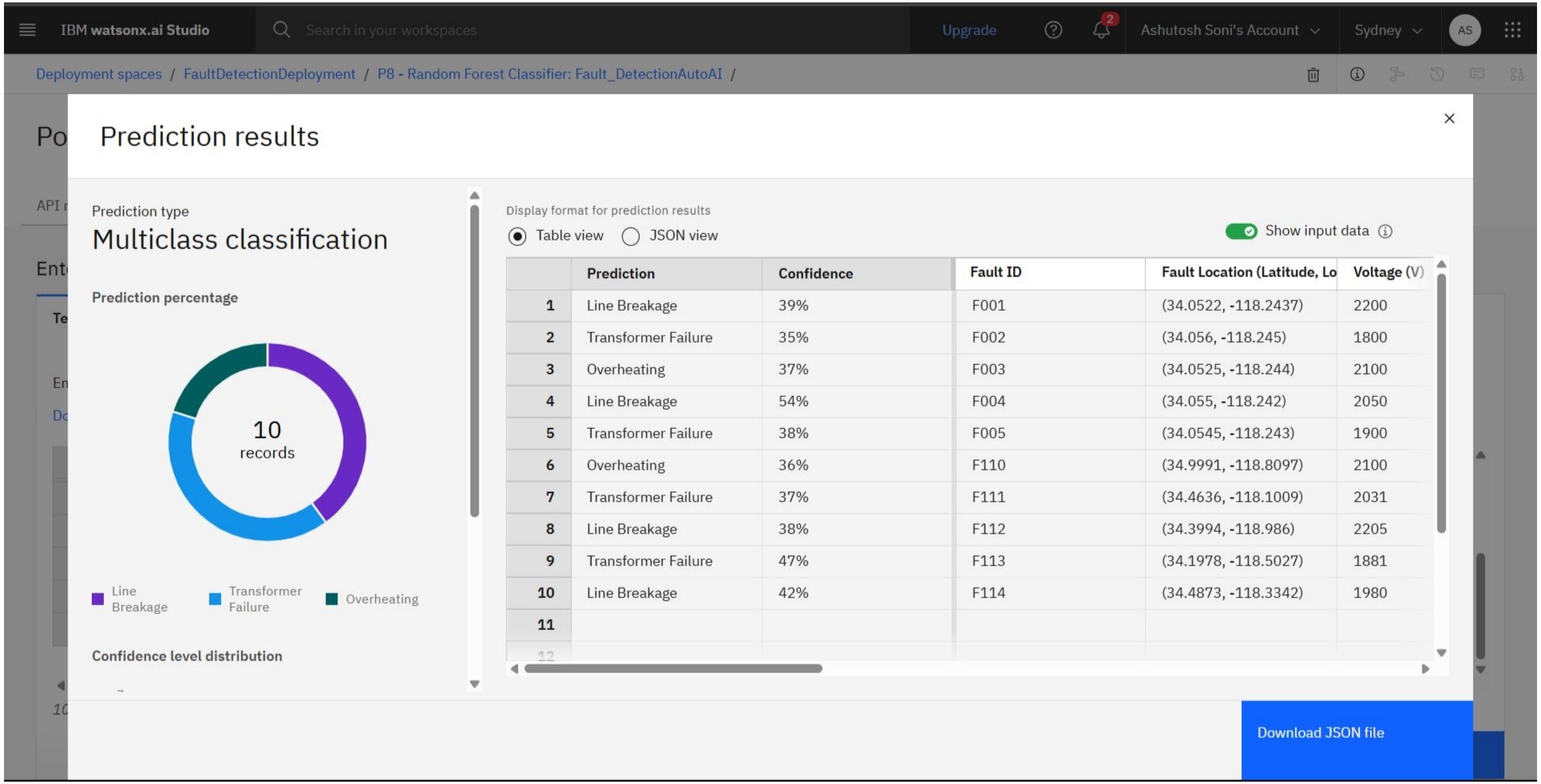


IBM watsonx.ai Studio
Search in your workspaces
Upgrade
2
Ashutosh Soni's Account
Sydney
AS
Deployment spaces / FaultDetectionDeployment / P8 - Random Forest Classifier: Fault_DetectionAutoAI /
PowerSystemFaultDetectionDeployment
Deployed
Online
API reference
Test
Enter input data
Text
JSON
Enter data 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

	Fault ID (other)	Fault Location (Latitude, Longitude) (other)	Voltage (V) (double)	Current (A) (double)	Power Load (MW) (double)	Temperature (°C) (double)	Wind Speed (km/h)
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	F004	(34.055, -118.242)	2050	240	48	23	10
5	F005	(34.0545, -118.243)	1900	190	50	30	18
6	F110	(34.9991, -118.8097)	2100	201	45	31	23

10 rows, 12 columns
Predict

RESULT



	Prediction	Confidence	Fault ID	Fault Location (Latitude, Lo	Voltage (V)
1	Line Breakage	39%	F001	(34.0522, -118.2437)	2200
2	Transformer Failure	35%	F002	(34.056, -118.245)	1800
3	Overheating	37%	F003	(34.0525, -118.244)	2100
4	Line Breakage	54%	F004	(34.055, -118.242)	2050
5	Transformer Failure	38%	F005	(34.0545, -118.243)	1900
6	Overheating	36%	F110	(34.9991, -118.8097)	2100
7	Transformer Failure	37%	F111	(34.4636, -118.1009)	2031
8	Line Breakage	38%	F112	(34.3994, -118.986)	2205
9	Transformer Failure	47%	F113	(34.1978, -118.5027)	1881
10	Line Breakage	42%	F114	(34.4873, -118.3342)	1980
11					

CONCLUSION

- This project demonstrates the effectiveness of machine learning in detecting and classifying power system faults using electrical measurement data. By accurately identifying fault types such as line-to-ground, line-to-line, and three-phase faults, the model enhances grid stability and enables faster response times. With potential for real-time deployment and integration into smart grid systems, the solution contributes to a more reliable and intelligent power distribution network.

FUTURE SCOPE

- 1. *Real-Time Grid Deployment*
-     Integrate the model with SCADA/PMU systems for real-time fault detection and response.
- 2. *Advanced Fault Handling*
-     Extend detection to complex, multi-fault, and intermittent fault scenarios.
- 3. *Scalable Transfer Learning*
-     Adapt models across different regions or grids using transfer learning.
- 4. *Smart Grid Integration*
-     Embed the system in IoT-enabled smart grids and edge devices for predictive maintenance.
- 5. *Explainable AI*
-     Use interpretable models (e.g., SHAP) to help operators understand fault causes.
- 6. *Cyber-Physical Security*

    Enhance fault models to detect and differentiate cyber-induced faults.
- 7. *Cloud-Based Scalability*
-     Deploy via IBM Cloud Lite and edge platforms for accessible, scalable monitoring.

REFERENCES

- Ziya Uddin, Power System Faults Dataset, Kaggle (https://www.kaggle.com/datasets/ziya07/power-system-faults-dataset)
- IBM Watsonx.ai Documentation (https://www.ibm.com/docs/en/watsonx)
- IEEE Xplore – Research articles on fault classification in power systems (https://ieeexplore.ieee.org)
- Scikit-learn: Random Forest & SVM Theory (https://scikit-learn.org/stable/user_guide.html)

edunet foundation

IBM CERTIFICATIONS


In recognition of the commitment to achieve professional excellence

Getting Started with Artificial Intelligence

IBM SkillsBuild

IBM

Ashutosh Kumar Soni

Has successfully satisfied the requirements for:

Getting Started with Artificial Intelligence

Issued on: Jul 15, 2025

Issued by: IBM SkillsBuild

Verify: https://www.credly.com/badges/2d66f096-3c03-4b5a-b8fb-c0e3f86db160

IBM

edunet foundation

IBM CERTIFICATIONS


Journey to Cloud: Envisioning Your Solution

IBM SkillsBuild

IBM

In recognition of the commitment to achieve professional excellence

Ashutosh Kumar Soni

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/d8402b45-323c-4979-9632-d5a94f131e16

IBM

IBM CERTIFICATIONS

IBM **SkillsBuild** Completion Certificate

This certificate is presented to

Ashutosh Kumar Soni

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