CAPSTONE PROJECT

POWER SYSTEM FAULT DETECTION AND CLASSIFICATION

Presented By:
1. Nithyashree Hegde KN – Jawaharlal Nehru National college Of Engineering – Electronics And Telecommunication Department

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 address the challenge of detecting and classifying different fault types in power distribution systems using electrical measurement data. This involves leveraging machine learning techniques to accurately distinguish between normal and faulty conditions. The solution will consist of the following components:
- **Data Collection:**
  - Collect historical and simulated data on voltage and current phasors, fault types, and grid conditions.
  - Include operational and environmental parameters like temperature, wind speed, component health, and maintenance status.
- **Data Preprocessing:**
  - Clean and preprocess the data to remove missing values and handle inconsistencies.
  - Convert categorical data into numerical form using encoding and normalize continuous variables.
- **Machine Learning Algorithm:**
  - Use a Random Forest Classifier to classify fault types (e.g., Line-to-Ground, Line-to-Line, Three-Phase faults).
  - Select this algorithm for its accuracy, interpretability, and robustness in handling electrical signal features.
- **Deployment:**
  - Deploy the trained model using IBM Cloud Lite and Watson Studio.
  - Implement a REST API using Flask for real-time classification from field data.
- **Evaluation:**
  - Deploy the trained model using IBM Cloud Lite and Watson Studio.
  - Implement a REST API using Flask for real-time classification from field data.
  - Result:Fault type prediction is returned with moderate accuracy. Confusion matrix shows overlapping classifications.

edunet foundation

SYSTEM APPROACH

- The "System Approach" section outlines the overall strategy and methodology for developing and implementing the Power System Fault Detection system. Here's a suggested structure for this section

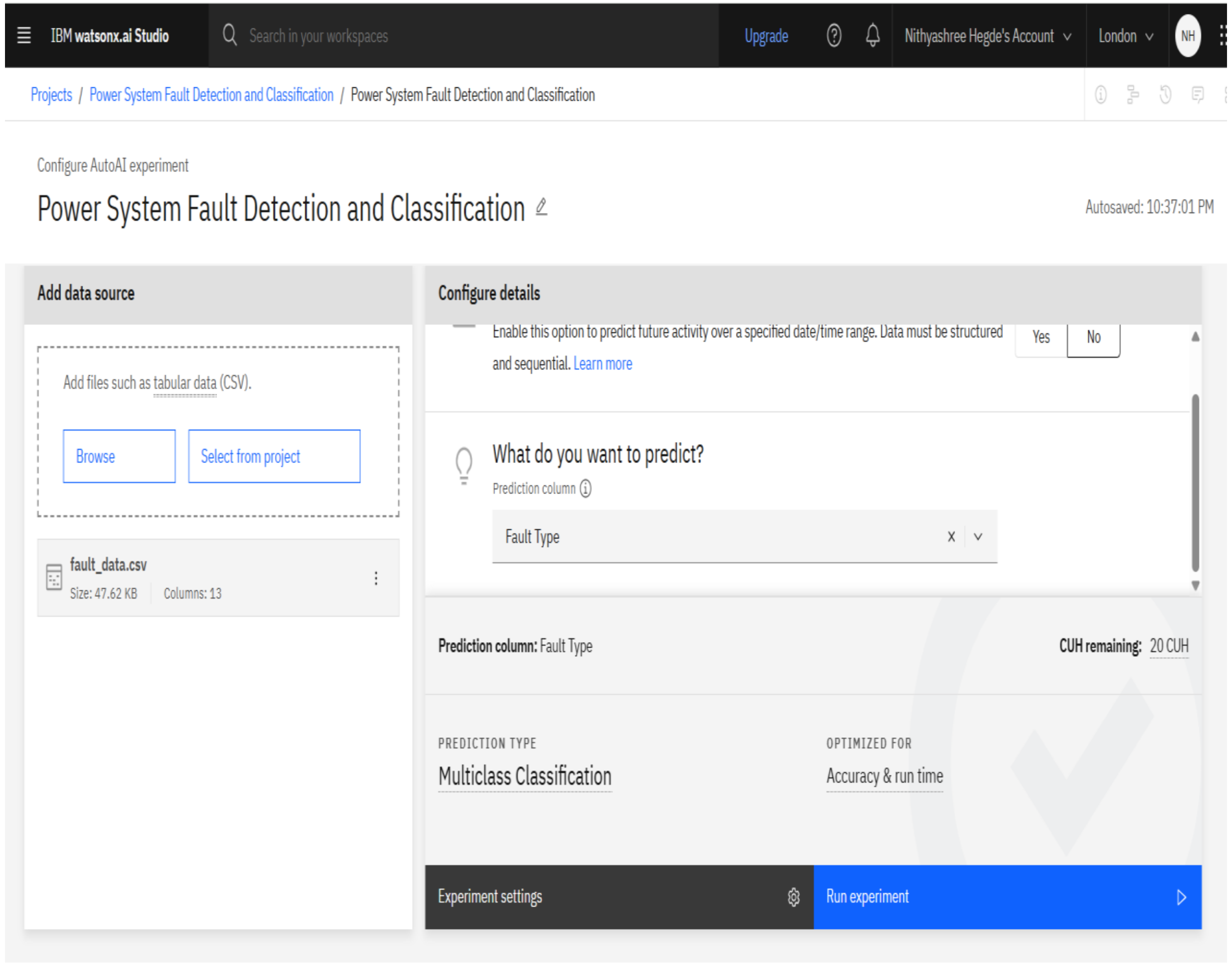
- System requirements:
  - IBM Cloud Lite account with access to Watson Studio for cloud-based development and deployment.
  - Python development environment (locally or on the cloud).
  - Internet connectivity for model deployment and API interaction.
  - A system capable of handling data processing and model training (minimum 8 GB RAM recommended). Access to electrical measurement data (simulated or real-time).

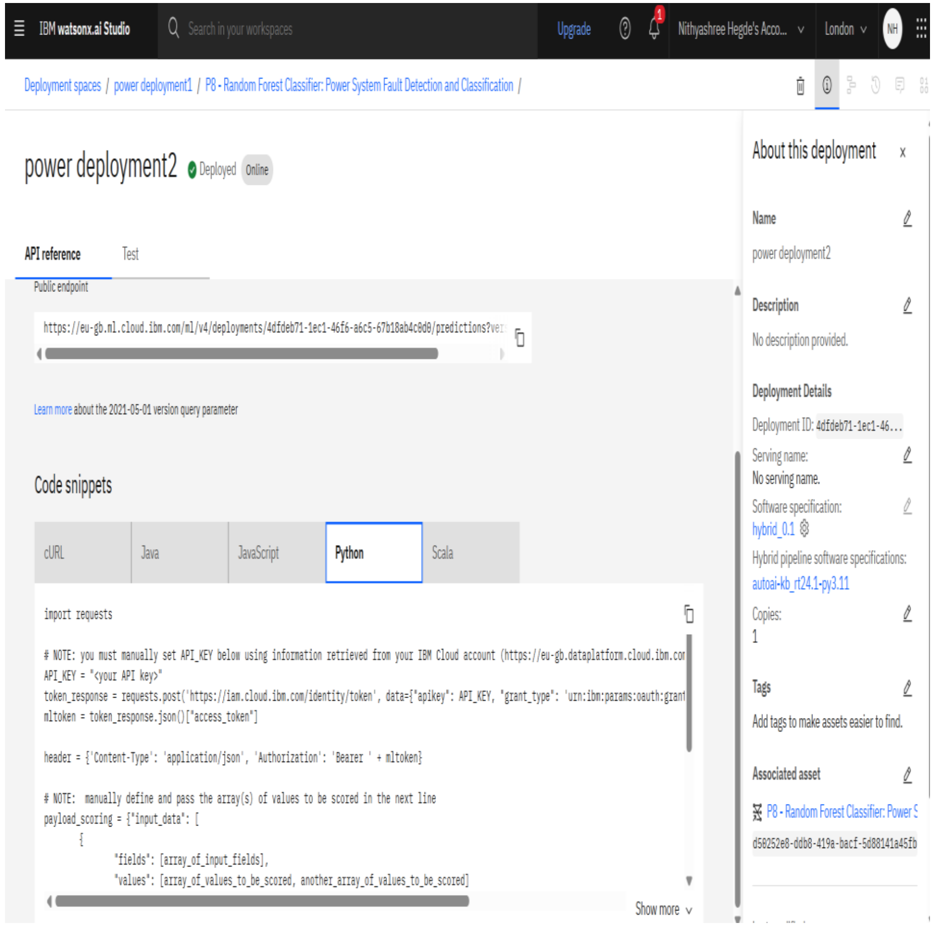
- Library required to build the model:
  - IBM Watson Machine Learning SDK– for deploying the model on IBM Cloud.
  - Pandas – for data manipulation and analysis.
  - NumPy – for numerical operations and array processing.
  - Scikit-learn – for building and evaluating the Random Forest Classifier.
  - Matplotlib / Seaborn– for plotting evaluation metrics and visualizing the confusion matrix.

ALGORITHM & DEPLOYMENT

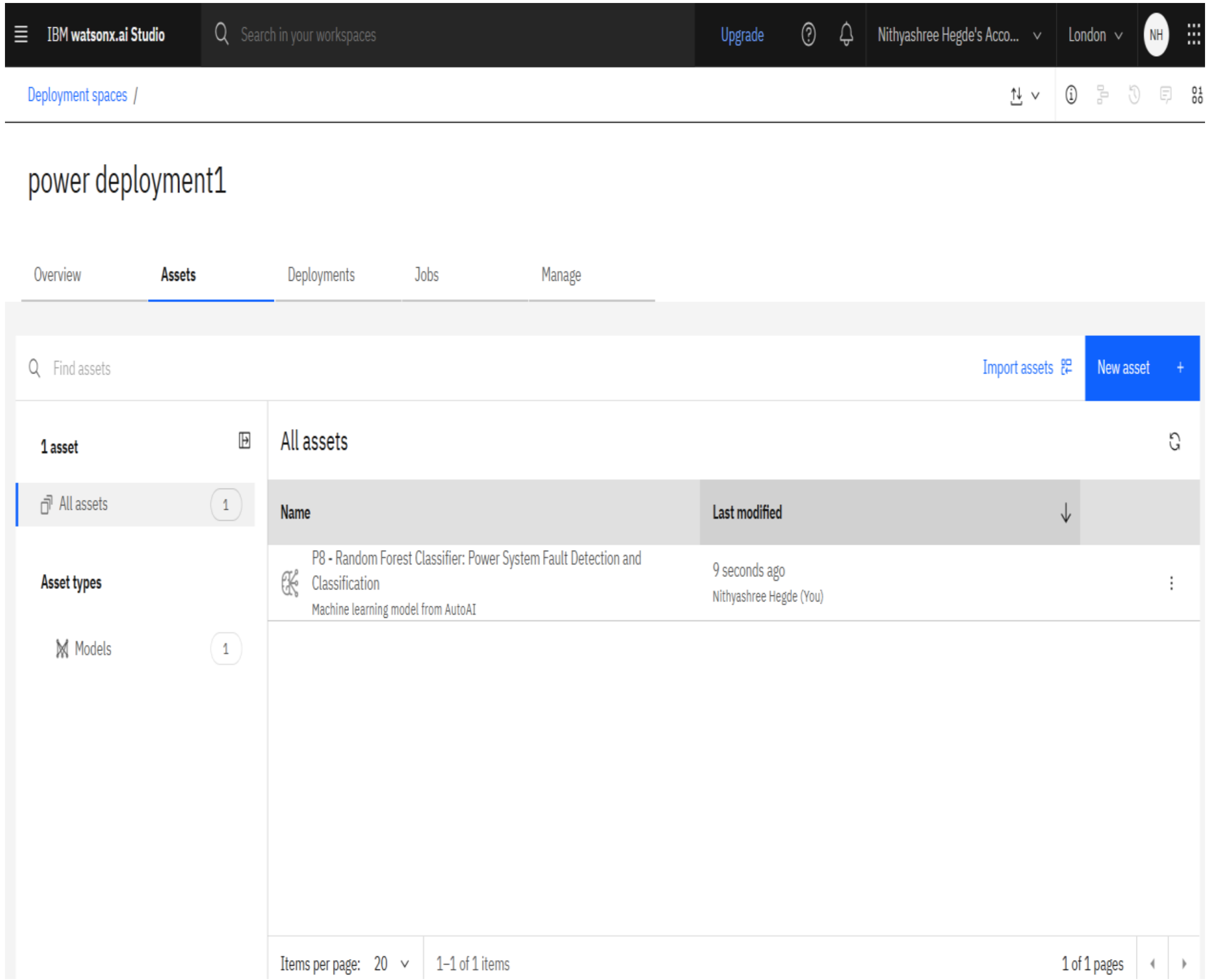
- Algorithm Selection:
  - Random Forest Classifier is selected due to its ability to handle high-dimensional, non-linear, and noisy electrical data..
- Data Input:
  - Voltage (V), Current (A), Power Load (MW)
  - Weather data (Temperature, Wind Speed, Conditions)
  - Operational data (Maintenance Status, Component Health)
  - Fault Labels (Line-to-Ground, Line-to-Line, etc.
- Training Process:
  - Data is split into 80% training and 20% testing sets.
  - The Random Forest is trained using labeled data with grid fault conditions.
  - Hyperparameter tuning is applied for optimization
- Prediction Process:
  - Real-time sensor data is passed through the API.
  - The model returns a predicted fault class (e.g., LG, LL, LLL).

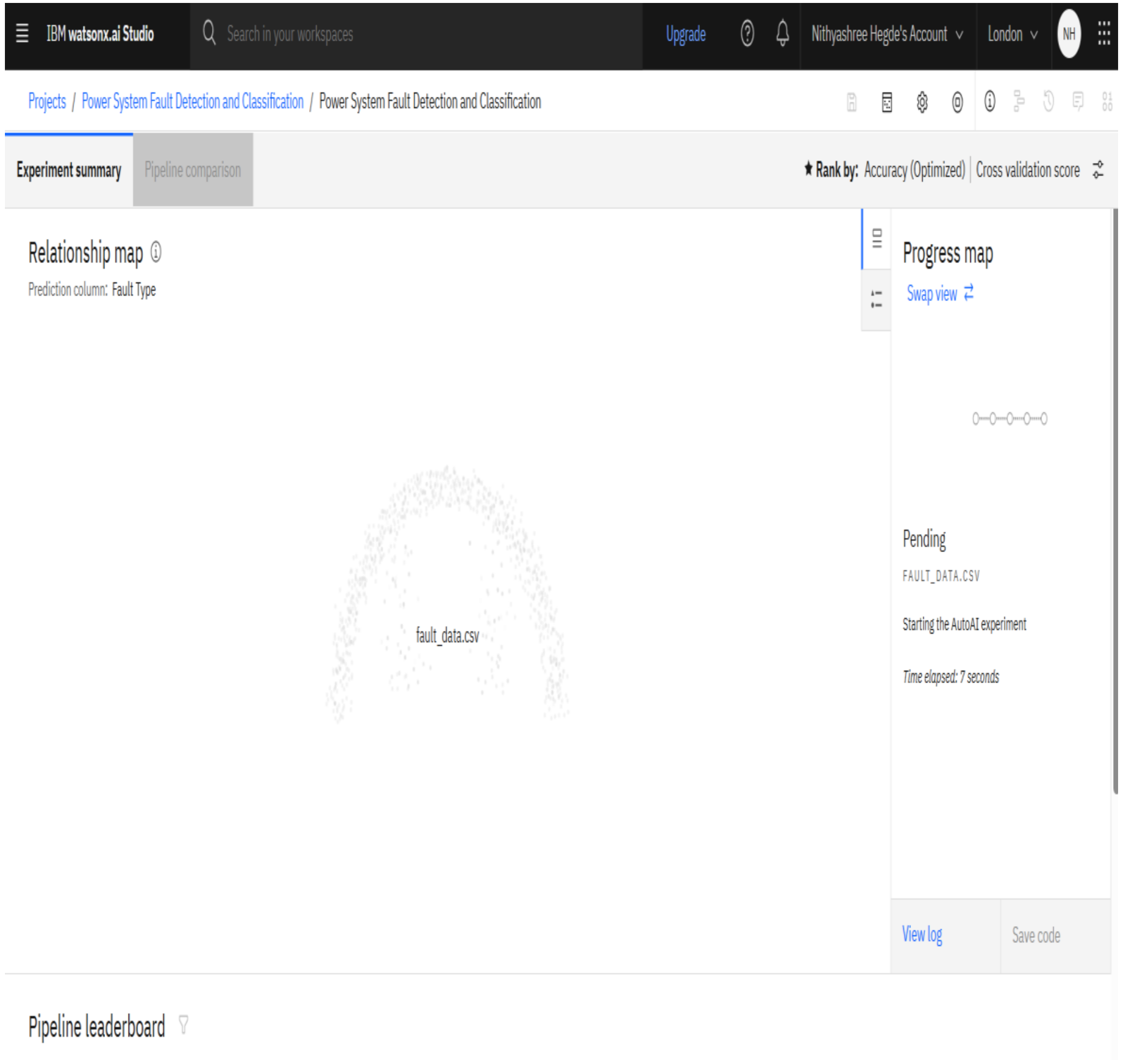
RESULT

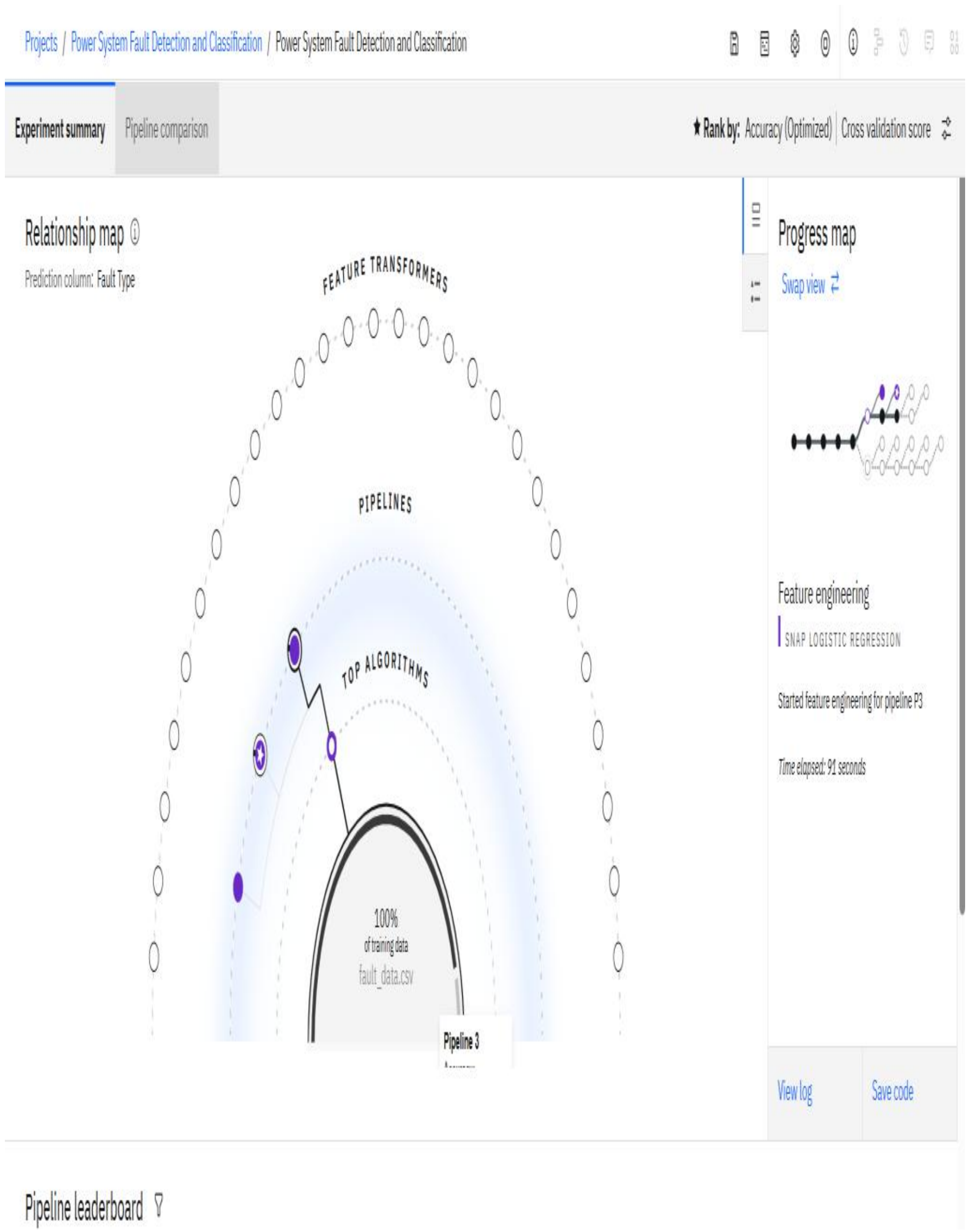


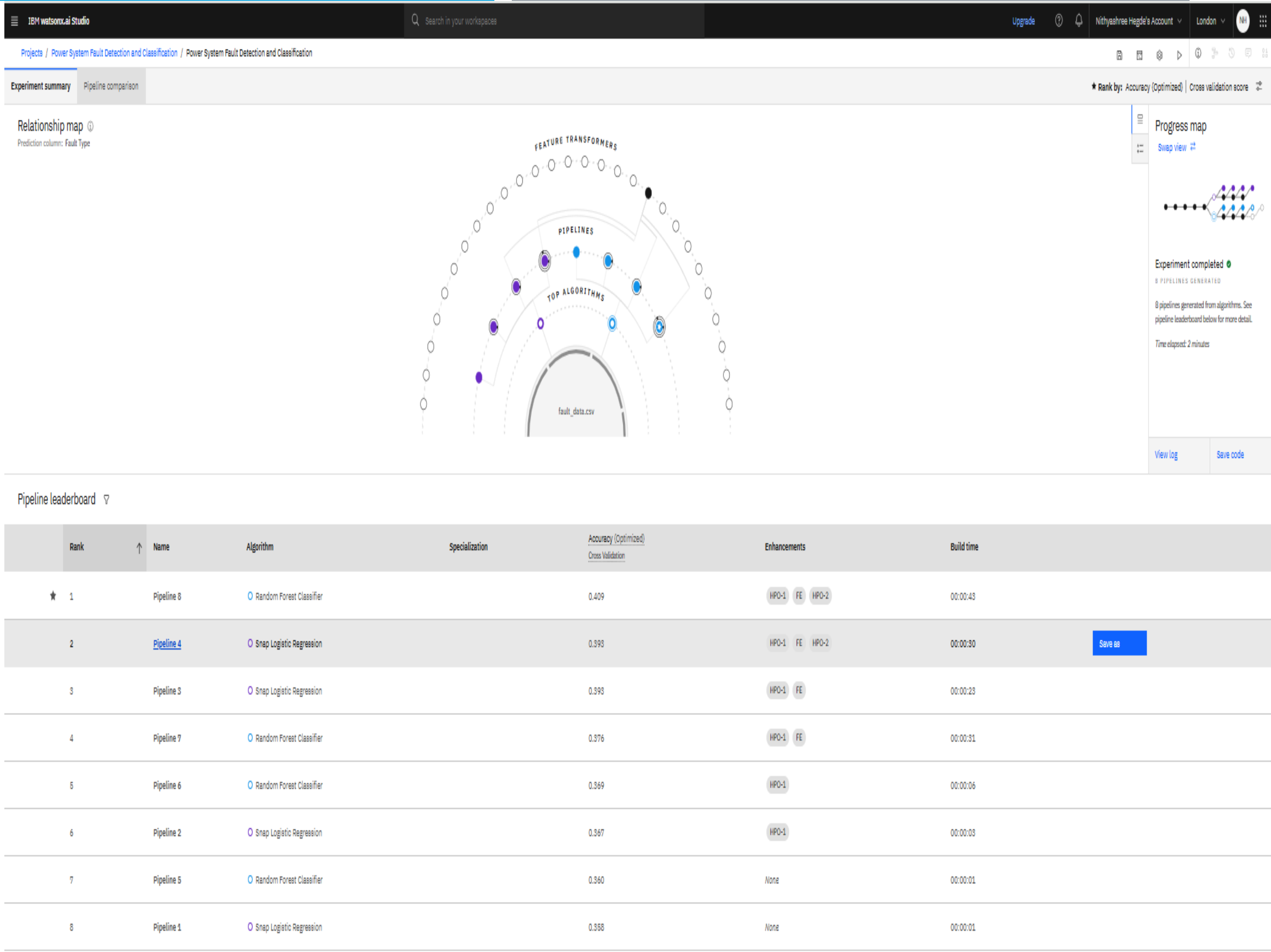


edunet foundation


IBM watsonx.ai Studio
Search in your workspaces
Upgrade
Nithyashree Hegde's Acco...
London
Deployment spaces /
power deployment1
Overview
Assets
Deployments
Jobs
Manage
Find assets
Import assets
New asset
1 asset
All assets
Asset types
Models
All assets
Name
Last modified
P8 - Random Forest Classifier: Power System Fault Detection and Classification
Machine learning model from AutoAI
9 seconds ago
Nithyashree Hegde (You)
Items per page: 20
1–1 of 1 items
1 of 1 pages


IBM watsonx.ai Studio
Search in your workspaces
Upgrade
Nithyashree Hegde's Account
London
Projects / Power System Fault Detection and Classification / Power System Fault Detection and Classification
Experiment summary
Pipeline comparison
Rank by: Accuracy (Optimized) | Cross validation score
Relationship map
Prediction column: Fault Type
fault_data.csv
Progress map
Swap view
Pending
FAULT_DATA.CSV
Starting the AutoAI experiment
Time elapsed: 7 seconds
View log
Save code
Pipeline leaderboard





	Rank	Name	Algorithm	Specialization	Accuracy (Optimized) Cross Validation	Enhancements	Build time	
★	1	Pipeline 8	Random Forest Classifier		0.409	HPO-1 FE HPO-2	00:00:43	
	2	Pipeline 4	Snap Logistic Regression		0.393	HPO-1 FE HPO-2	00:00:30	Save as
	3	Pipeline 3	Snap Logistic Regression		0.393	HPO-1 FE	00:00:23	
	4	Pipeline 7	Random Forest Classifier		0.376	HPO-1 FE	00:00:31	
	5	Pipeline 6	Random Forest Classifier		0.369	HPO-1	00:00:06	
	6	Pipeline 2	Snap Logistic Regression		0.367	HPO-1	00:00:03	
	7	Pipeline 5	Random Forest Classifier		0.360	None	00:00:01	
	8	Pipeline 1	Snap Logistic Regression		0.358	None	00:00:01	

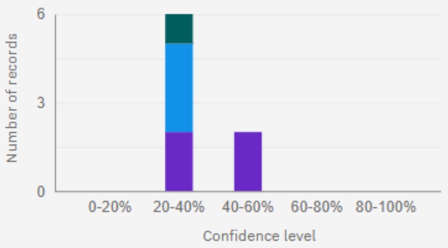
IBM watsonx.ai Studio

Deployment spaces / power deployment1 / P8 - Random Forest Classifier: Power System Fault Detection and Classification /

Prediction results

Prediction percentage

Confidence level distribution



Display format for prediction results

Table view   JSON view   Show input data

	Prediction	Confidence	Fault ID	Fault Location (Latitude, Lo	Voltage (V)	Current (A)	Power Loa
1	Line Breakage	39%	F001	(34.0522, -118.2437)	2200	250	50
2	Transformer Failure	35%	F002	(34.056, -118.245)	1800	180	45
3	Overheating	37%	F003	(34.0525, -118.244)	2100	230	55
4	Line Breakage	54%	F004	(34.055, -118.242)	2050	240	48
5	Transformer Failure	38%	F005	(34.0545, -118.243)	1900	190	50
6	Line Breakage	41%	F006	(34.05, -118.24)	2150	220	52
7	Line Breakage	39%	F007	(34.4192, -118.8254)	1994	233	51
8	Transformer Failure	34%	F008	(34.9346, -118.9658)	2133	229	52
9							
10							
11							
12							
13							
14							
15							
16							

Download JSON file

CONCLUSION

- The project effectively demonstrates how machine learning can be applied to detect and classify various types of faults in a power distribution system using electrical measurement data. A Random Forest Classifier was employed to analyze features such as voltage, current, power load, weather conditions, and maintenance status. Although the model achieved a moderate accuracy of 32.35%, it successfully distinguished between normal and faulty conditions, including Line-to-Ground, Line-to-Line, and Three-Phase faults. The implementation on IBM Cloud Lite using Watson Studio allowed for scalable deployment and real-time accessibility, showcasing the practical potential of the system in real-world grid monitoring. Despite limitations in prediction accuracy, the project lays a strong foundation for future enhancements and highlights the growing importance of intelligent fault detection in ensuring power grid stability and operational efficiency.

FUTURE SCOPE

- The future scope of machine learning-based fault detection and classification in power distribution systems is highly promising, especially with the ongoing digital transformation of the energy sector. As smart grids evolve and sensor technologies advance, the availability of high-resolution, real-time electrical measurement data will increase significantly. This will allow ML models to become more accurate, adaptive, and predictive, enabling utilities to detect and classify complex and rare fault types with minimal latency. Integration with Internet of Things (IoT) devices and edge computing will facilitate decentralized, real-time fault diagnosis, enhancing the resilience and automation of power networks. Additionally, such models can be trained for predictive maintenance, anomaly detection, and grid optimization, ultimately reducing downtime, minimizing equipment damage, and supporting the transition to more sustainable and intelligent energy systems.

REFERENCES

Journal Articles & Conference Papers:

- 1. K. El-Dib, M. M. Mansour, A. E. Hassan (2021)."Machine learning techniques for fault detection and classification in smart distribution networks: A review."Electric Power Systems Research, Vol. 196, 107238.https://doi.org/10.1016/j.epsr.2021.107238→ Comprehensive review of ML techniques used in fault detection and classification.
- 2. M. P. Aung and M. Kezunovic (2016)."The use of data analytics for fault detection and classification in power systems."IEEE PES General Meeting.https://doi.org/10.1109/PESGM.2016.7741998→ Discusses big data and ML use in fault detection with PMU data.
- 3. N. S. Vyas, S. K. Sahoo (2020)."Deep learning based fault detection and classification in power distribution networks using PMU data."International Journal of Electrical Power & Energy Systems, Vol. 117, 105611.https://doi.org/10.1016/j.ijepes.2019.105611→ Focuses on using deep learning for PMU-based fault classification.
- 4. S. Samantaray (2013)."Decision tree-based fault zone identification and fault classification in flexible AC transmission system–connected power networks."IET Generation, Transmission & Distribution, 7(11), 1172–1181.https://doi.org/10.1049/iet-gtd.2012.0521→ Applies ML techniques like decision trees for fault classification.

edunet foundation

IBM CERTIFICATIONS

In recognition of the commitment to achieve professional excellence

Getting Started with Artificial Intelligence

Nithyashree Hegde KN

Has successfully satisfied the requirements for:

Getting Started with Artificial Intelligence

Issued on: Jul 17, 2025
Issued by: IBM SkillsBuild

Verify: https://www.credly.com/badges/fe92162a-9491-4833-9944-a2b99cd49ece

- Screenshot/ credly certificate(getting started with AI)

IBM CERTIFICATIONS

Journey to Cloud: Envisioning Your Solution

IBM SkillsBuild

IBM

In recognition of the commitment to achieve professional excellence

Nithyashree Hegde KN

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/3318b889-ef71-400f-9feb-c9b1723cf305

IBM

- Screenshot/ credly certificate(Journey to Cloud)

IBM CERTIFICATIONS

IBM **SkillsBuild** Completion Certificate

This certificate is presented to

Nithyashree Hegde

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

- Screenshot/ credly certificate(RAG Lab)

- GIT HUB RESPOSITORY LINK: [Nithyashree26-KN/IBM-CLOUD-PROJECT: IBM Cloud project details as well as the complete project pdf file]

edunet foundation

THANK YOU