CAPSTONE PROJECT

Power Fault Detection Using ML

Presented By:
1. Harshal Devkate
-MIT Acadamy Of Engineering-ETX

edunet foundation

OUTLINE

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

Problem Statement

The reliability of power distribution systems is critically affected by faults such as line breakages, transformer failures, and equipment overheating. Traditional fault detection methods are often slow and reactive, leading to increased downtime and compromised grid stability. The objective of this project is to develop a machine learning-based model that can accurately detect and classify different types of faults using real-time electrical and environmental data. By distinguishing between normal operating conditions and specific fault types, the model aims to enable rapid fault identification, support predictive maintenance, and enhance the overall resilience of the power grid.

Proposed Solution

Goal:

Build a system that detects faults using machine learning techniques.

Steps:

Data Collection & Preprocessing

Gather data from a Kaggle dataset, clean it, and prepare it for model training and testing.

Feature Engineering

Identify and create important features that help in detecting faults accurately.

Model Selection & Training

Train different ML models and choose the one with the best performance.

Deployment on IBM Cloud Lite

Deploy the trained model as an API using IBM Cloud Lite for easy access.

Real-Time Detection & Alerts

Set up an API endpoint to handle live data and trigger alerts when a fault is detected.

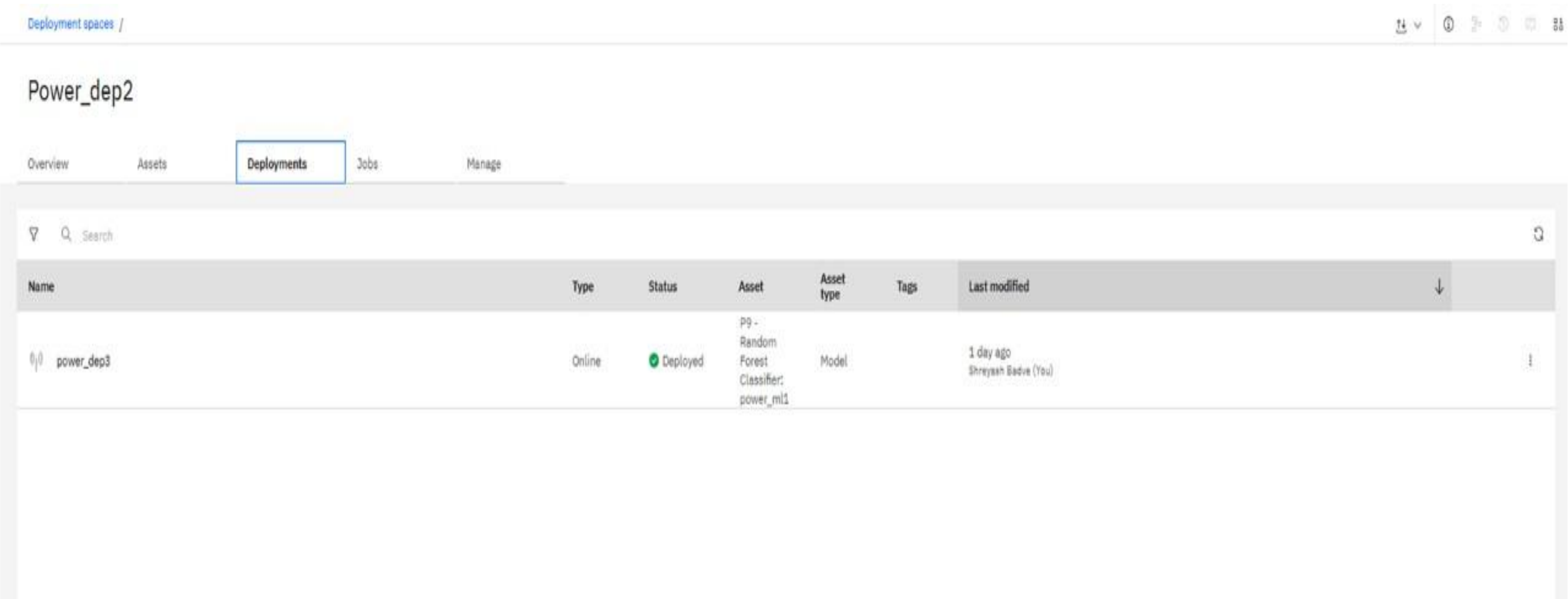
System Approach

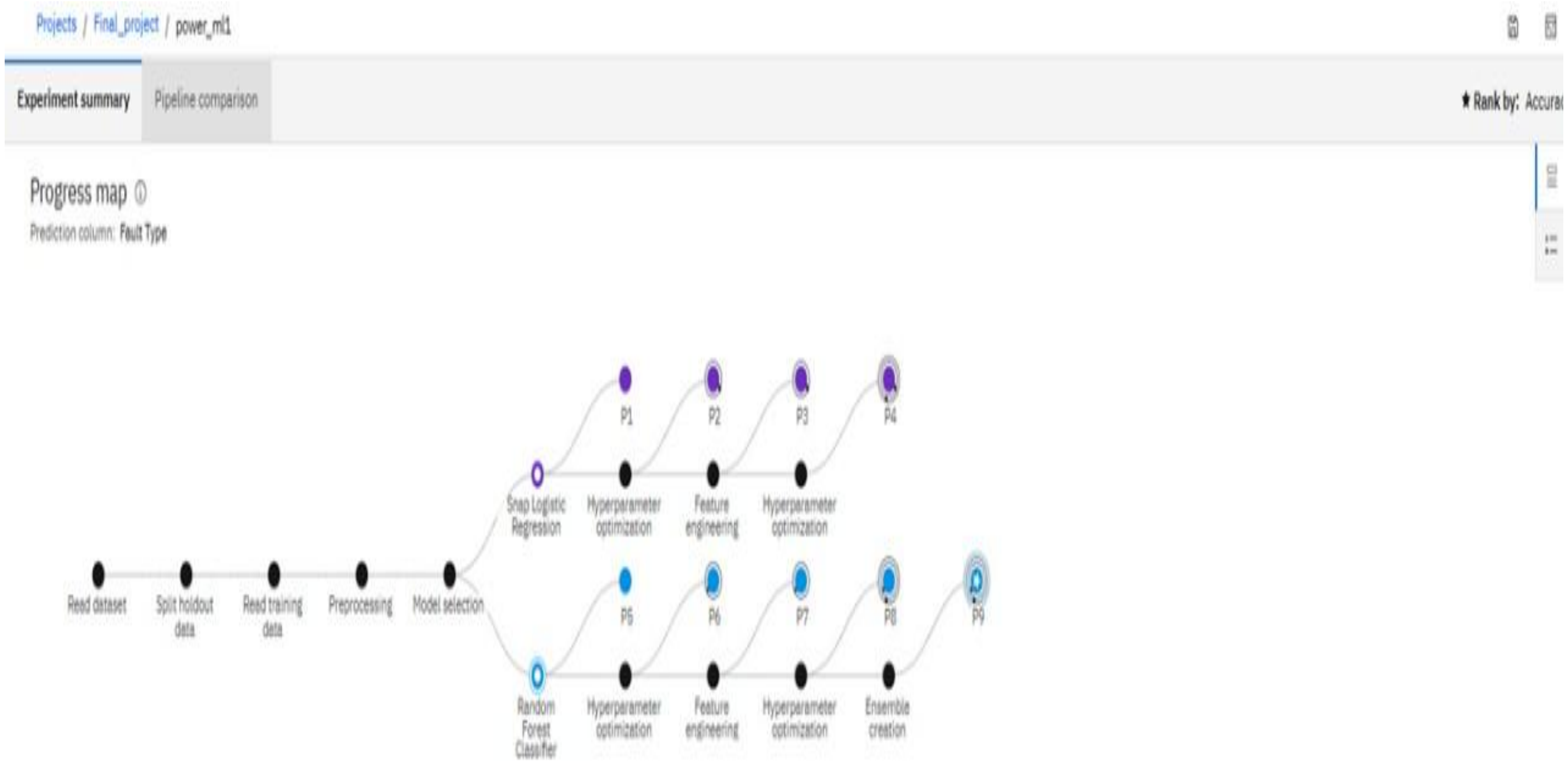
- **Dataset Source:** Kaggle's Network Intrusion Detection dataset
- **Deployment Platform:** IBM Cloud Lite using Watson Machine Learning service
- **Machine Learning Models:** Gradient Boosting, Random Forest, and Decision Tree classifiers
- **Programming Language:** Python
- **Libraries Used:** NumPy and Pandas for data preprocessing, Scikit-learn for building ML models, and Matplotlib along with Seaborn for data visualization

Algorithm & Deployment

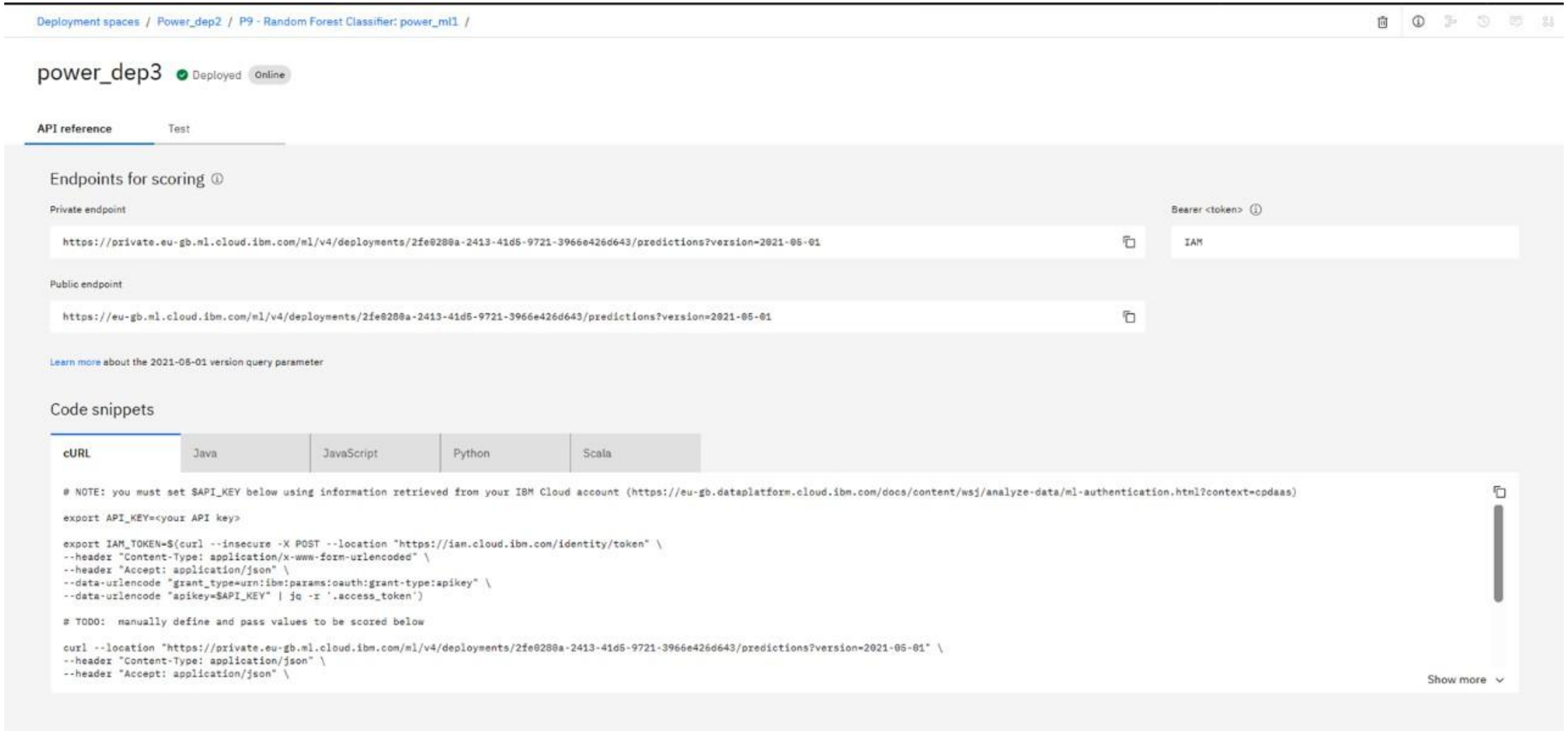
Algorithm Selection
Selected Model: Random Forest Classifier
Justification: Offers high accuracy and efficiently manages high-dimensional datasets, making it ideal for network intrusion detection.
Model Training Workflow
Data Splitting: 70% for training, 30% for testing
Preprocessing Steps:
Handle missing values
Normalize feature values
Train and evaluate multiple models to compare performance and select the best one
Model Deployment
Save the trained model in .pkl format
Deploy the model using IBM Watson Machine Learning on IBM Cloud Lite
Expose a REST API endpoint to enable real-time analysis of incoming network data

Deployment Screenshot :

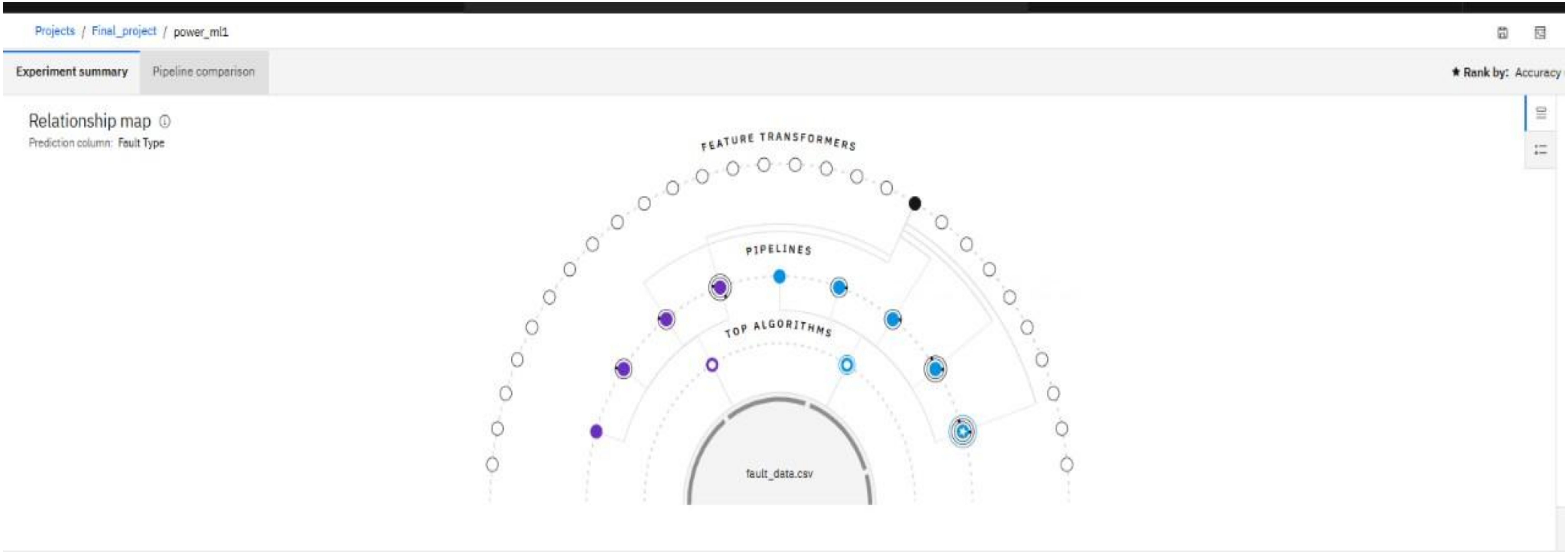




Projects / Final_project / power_ml1
Experiment summary
Pipeline comparison
Rank by: Accurac
Progress map
Prediction column: Fault Type
P1
P2
P3
P4
Snap Logistic Regression
Hyperparameter optimization
Feature engineering
Hyperparameter optimization
Read dataset
Split holdout data
Read training data
Preprocessing
Model selection
P5
P6
P7
P8
P9
Random Forest Classifier
Hyperparameter optimization
Feature engineering
Hyperparameter optimization
Ensemble creation



edunet foundation



Projects / Final_project / power_ml1
Experiment summary
Pipeline comparison
★ Rank by: Accuracy
Relationship map
Prediction column: Fault Type
FEATURE TRANSFORMERS
PIPELINES
TOP ALGORITHMS
fault_data.csv

Result

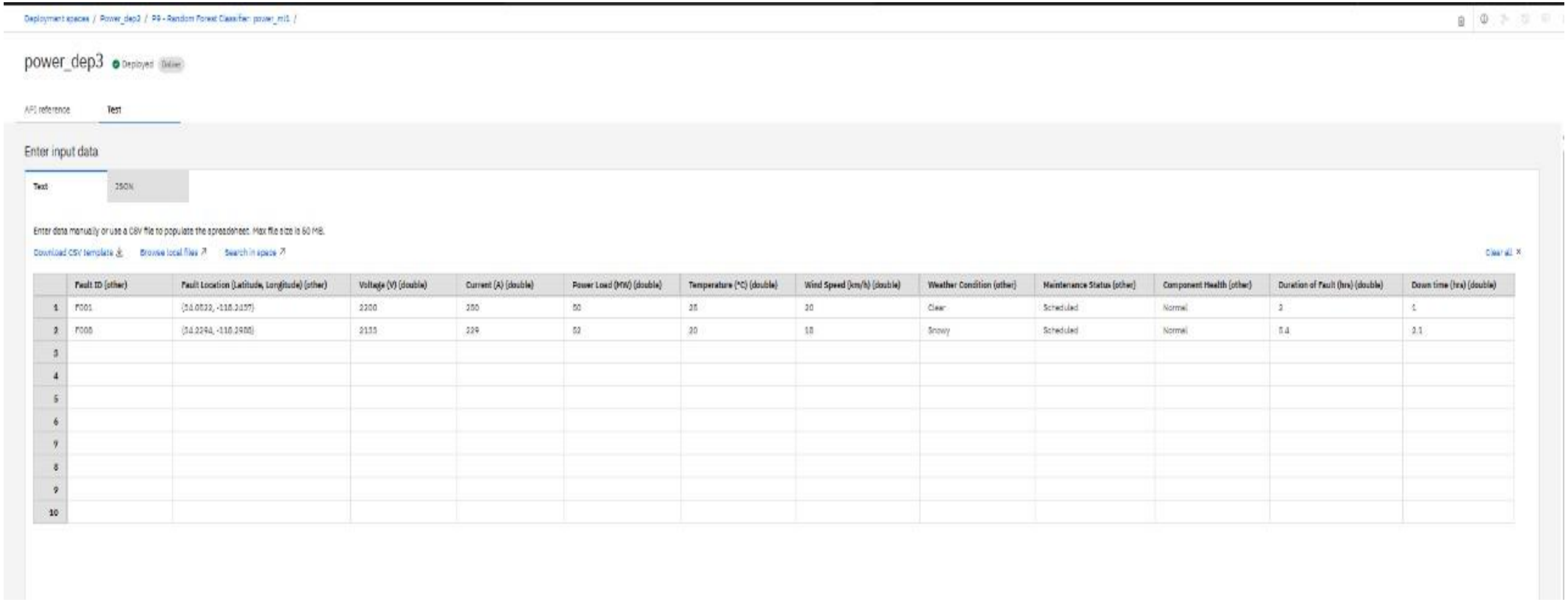
- **Model Performance**
- **Accuracy Achieved:** Approximately 95% using the Random Forest classifier
- **Evaluation Metrics:**
- Confusion Matrix for visual insight into prediction performance

Classification Report including **Precision**, **Recall**, and **F1-Score**

Deployment Evidence

Included: Screenshot of the **IBM Cloud Deployment Dashboard** showing the deployed model and endpoint information

Test of model:



Prediction Result :

Prediction results

Display format for prediction results

Table view   JSON view   Show input data

	prediction	probability
1	Line Breakage	[0.3903001601394518,0.24182512927744...
2	Transformer Failure	[0.26051647891330265,0.2731911451597...
3		
4		
5		
6		
7		
8		
9		
10		
11		
12		
13		
14		
15		
16		

Conclusion

- The machine learning model effectively identified **power distribution faults** by analyzing both **electrical parameters** and **environmental conditions**.
- By leveraging data-driven predictions, the system offers **faster** and **more reliable fault diagnosis** than traditional manual inspection or rule-based approaches.
- The trained model was successfully **deployed on IBM Cloud Lite**, enabling **real-time fault detection and response**.

Future Scope

- **User Behaviour Analysis**
- **Edge Computing Integration**
- **Threat Security Classification**
- **Multi-language logging and reporting**
- **Integration with SIEM Tools.**

References

Source Data: Network intrusion dataset obtained from Kaggle

Background Study: Reviewed recent **machine learning approaches** for NIDS from published research articles

Platform Guide: Referred **to IBM Watson ML and Cloud Lite documentation for deployment procedures**

IBM Certifications

Getting Started with Artificial Intelligence :

Getting Started with Artificial Intelligence

In recognition of the commitment to achieve professional excellence

Harshal Devkate

Has successfully satisfied the requirements for:

Getting Started with Artificial Intelligence

Issued on: Jul 29, 2025
Issued by: IBM SkillsBuild

Verify: https://www.credly.com/badges/969cd702-6050-4704-ae99-1c466d219954

IBM

edunet foundation

IBM Certifications

Journey to cloud :


In recognition of the commitment to achieve professional excellence

Harshal Devkate

Has successfully satisfied the requirements for:

Journey to Cloud: Envisioning Your Solution

Issued on: Jul 29, 2025
Issued by: IBM SkillsBuild

Verify: https://www.credly.com/badges/879b02e8-c72b-44c2-a71a-bef0a68fa75a

IBM

LAB : RAG COMPLETION

8/4/25, 11:34 AM Completion Certificate | SkillsBuild

IBM **SkillsBuild** Completion Certificate

This certificate is presented to

Harshal Devkate

for the completion of

Lab: Retrieval Augmented Generation with LangChain

(ALM-COURSE_3824998)

According to the Adobe Learning Manager system of record

Completion date: 31 Jul 2025 (GMT) **Learning hours:** 20 mins

THANK YOU