CAPSTONE PROJECT

POWER SYSTEM FAULT DETECTION AND CLASSIFICATION USING MACHINE LEARNING

Presented By:
1. SHREENIWAS PANDIT- COEP Technological University- Metallurgy and Materials Technology

edunet foundation

OUTLINE

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

PROBLEM STATEMENT

Modern power distribution systems are susceptible to various faults caused by factors like weather, equipment failure, or human error. Traditional detection methods can be slow and may not provide immediate insight into the nature of the fault, leading to prolonged outages and potential damage to the grid infrastructure. This project aims to address this critical gap by leveraging machine learning. The goal is to develop an intelligent system that continuously monitors electrical parameters like voltage and current. By training a model on datasets containing both normal and fault signatures, the system will learn to instantly detect an anomaly and precisely classify its type—whether it's a single line-to-ground fault or a more severe three-phase fault. The successful implementation of this model will lead to a significant reduction in fault response times, thereby enhancing grid reliability, improving safety, and minimizing the economic impact of power disruptions.

PROPOSED SOLUTION

- The proposed system aims to develop and deploy an intelligent fault detection model on the IBM Cloud platform. This solution will leverage IBM Watson AI services to analyze electrical data and provide real-time classifications of the power grid's status.
- **Data Collection**: Gather historical datasets of power system measurements, specifically voltage and current phasors, which include examples of both normal operation and various fault conditions.
- **Data Preprocessing**: Utilize tools within IBM Watson Studio to clean the data, handle any inconsistencies, and engineer features that help the model distinguish between different types of faults.
- **Machine Learning Algorithm**: Implement a robust classification algorithm within IBM Watson Studio. This could involve using the AutoAI feature to find the best model or manually building a model like a Random Forest, Gradient Boosting, or a Neural Network to classify the grid's state.
- **Deployment**: Deploy the trained model as a scalable web service using IBM Watson Machine Learning. This creates a REST API endpoint that can be queried in real-time by grid monitoring systems.
- **Evaluation**: Assess the model's performance using standard classification metrics like accuracy, precision, recall, and a confusion matrix to verify its effectiveness in identifying different fault types correctly.

SYSTEM APPROACH

- Cloud Platform: IBM Cloud
- AI/ML Services:
- IBM Watson Studio: For data preparation, model development, training, and evaluation.
- IBM Watson Machine Learning: For deploying the model as a production-ready API.
- Core Libraries/Language:

  Language: Python

  Data Handling: Pandas, NumPy

  ML Model: Scikit-learn

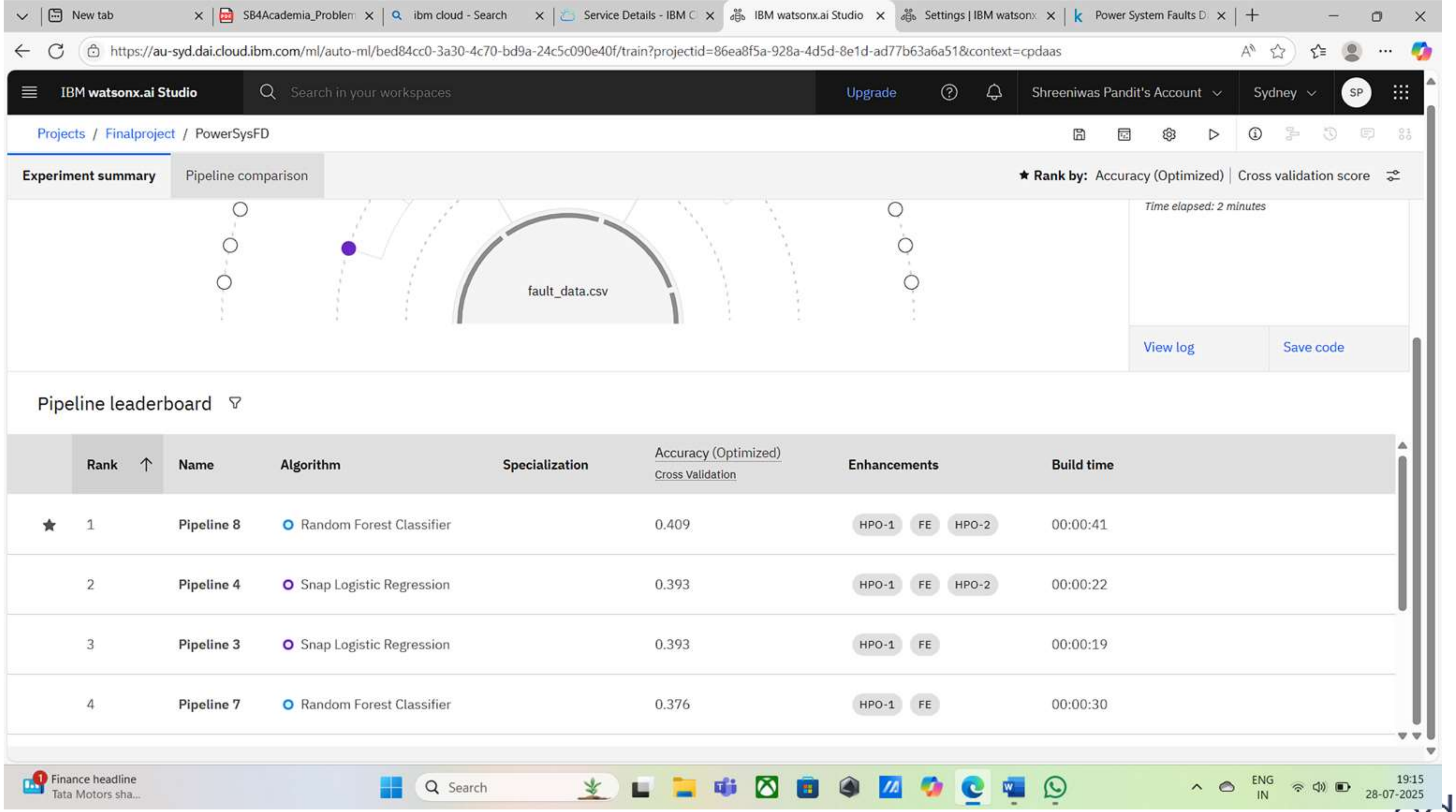
  Deployment Interaction: ibm-watson-machine-learning library

  Algorithm & Deployment.

ALGORITHM & DEPLOYMENT

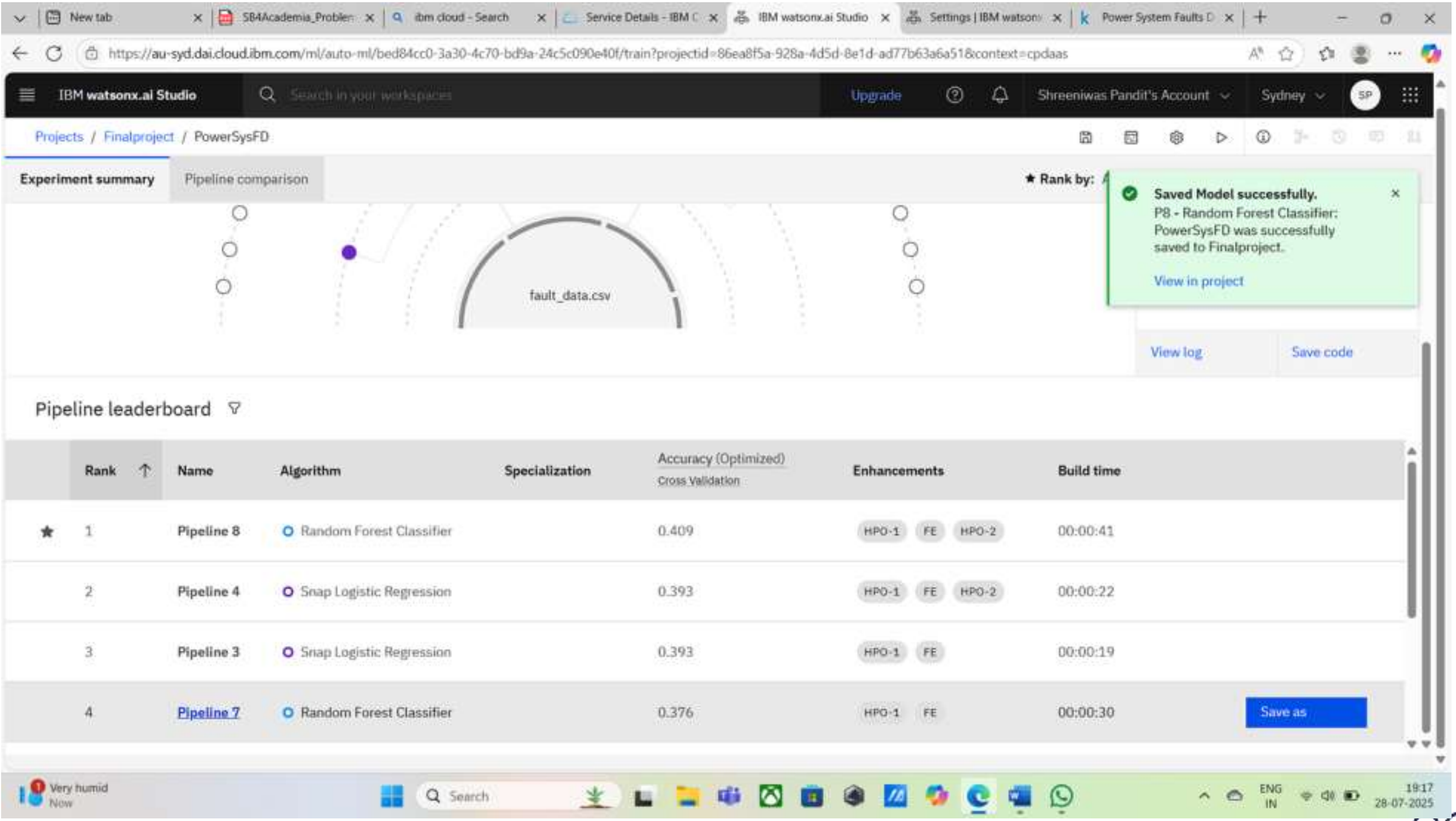
- **Algorithm Selection**: A Random Forest Classifier is chosen for this project. This algorithm is highly effective for classification tasks, is robust to overfitting, and can provide insights into which electrical measurements are most important for detecting a fault.
- **Data Input**: The model will be trained using input features from the power system data, such as the magnitude and angle of the three-phase voltages and currents and other data given in kaggle data.
- **Training Process**: Within Watson Studio, the historical dataset will be split into training and testing sets. The Random Forest model will be trained on the training data, and techniques like hyperparameter tuning will be applied to optimize its performance.
- **Deployment & Prediction**: The finalized model is deployed via IBM Watson Machine Learning. This provides a secure API endpoint. A monitoring system can send a JSON object with the latest voltage and current readings to this endpoint. The API will process this input and instantly return a JSON response with the predicted status

RESULT



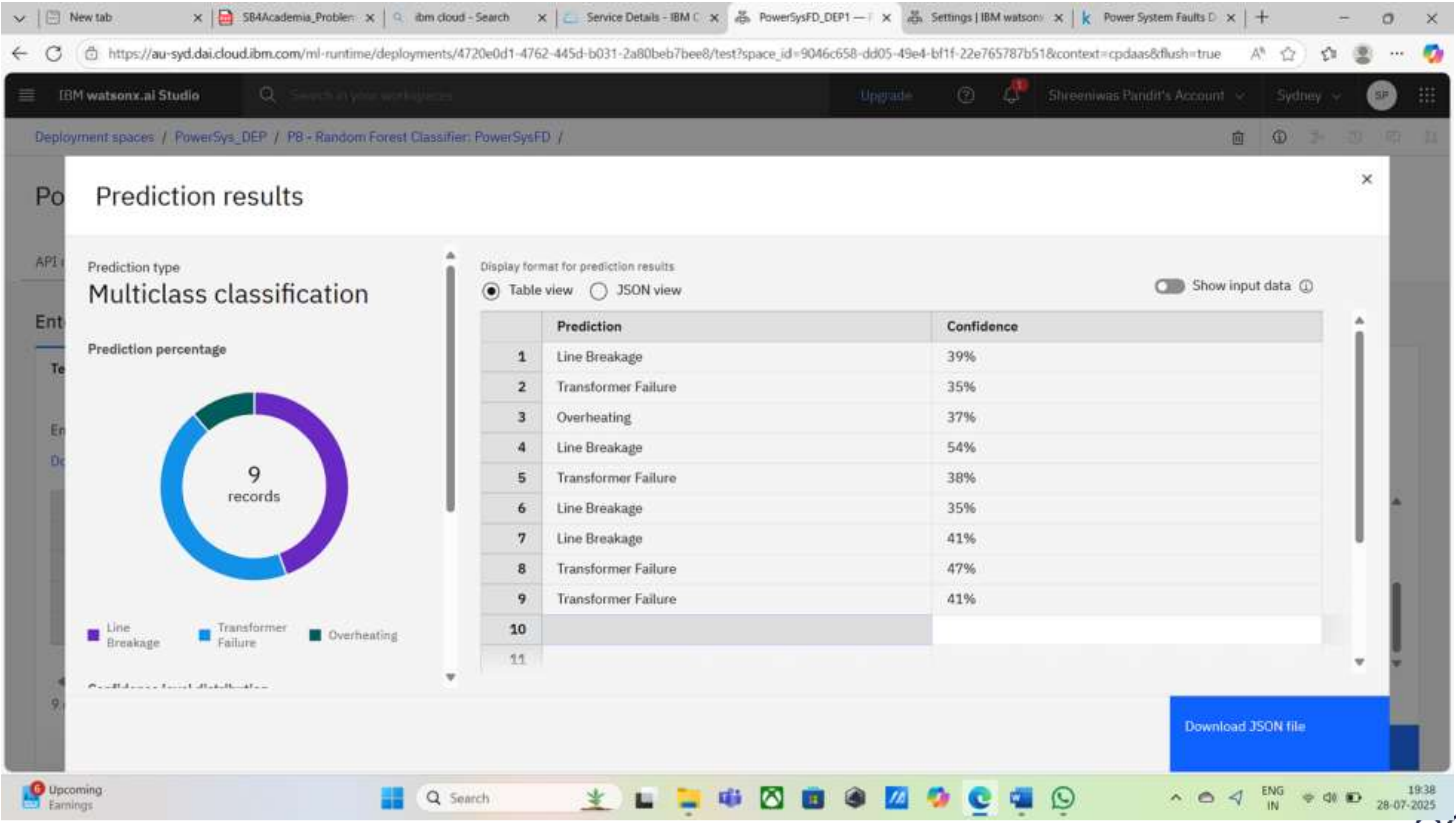
edunet foundation

RESULT

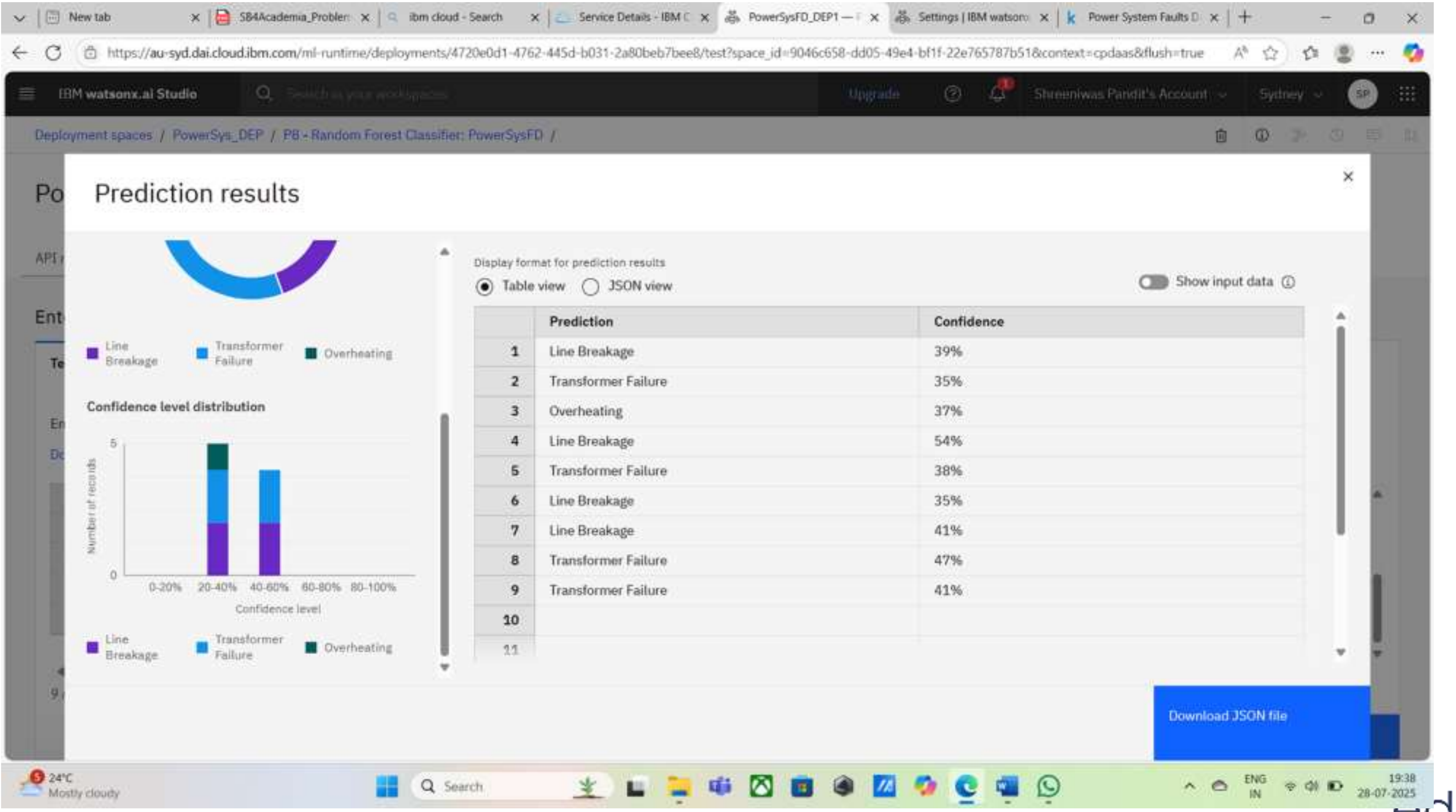


edunet foundation

RESULT


IBM watsonx.ai Studio
Deployment spaces / PowerSys_DEP / P8 - Random Forest Classifier: PowerSysFD /
Prediction results
Prediction type
Multiclass classification
Prediction percentage
9
records
Line Breakage
Transformer Failure
Overheating
Display format for prediction results
Table view
JSON view
Show input data
Prediction
Confidence
1 Line Breakage 39%
2 Transformer Failure 35%
3 Overheating 37%
4 Line Breakage 54%
5 Transformer Failure 38%
6 Line Breakage 35%
7 Line Breakage 41%
8 Transformer Failure 47%
9 Transformer Failure 41%
Download JSON file

RESULT



edunet foundation

CONCLUSION

- This project successfully demonstrated an end-to-end workflow for creating and deploying a power system fault detection model using IBM Watson Studio.
- The Watson AutoAI experiment automatically evaluated multiple algorithms, identifying "Pipeline 8," a Random Forest Classifier, as the top-performing model.
- The selected model achieved an optimized cross-validation accuracy of 0.409 (40.9%).
- The best model was successfully saved and deployed as a web service for real-time predictions.
- The deployed system is functional and capable of performing multiclass classification to predict faults like "Line Breakage," "Transformer Failure," and "Overheating," each with an associated confidence score.
- While the system is successfully deployed, the resulting accuracy indicates the complexity of the dataset and suggests that further model tuning or data enhancement is required for improved real-world performance.

FUTURE SCOPE

- **Enhanced Data Integration**: Incorporate additional data sources, such as real-time weather data or transformer temperature readings, to potentially predict faults before they occur.
- **Advanced Models:** Explore deep learning algorithms like Long Short-Term Memory (LSTM) networks within Watson Studio to better analyze the time-series nature of electrical data for improved accuracy.
- **Edge Deployment**: Consider integrating emerging technologies by deploying a lightweight version of the model on edge devices closer to the monitoring equipment to reduce latency and provide even faster responses.

REFERENCES

- IBM Watson Studio - Official Documentation

Link: https://www.ibm.com/products/watson-studio

- IBM AutoAI - Official Documentation

Link: https://www.ibm.com/docs/en/cloud-paks/cp-data/4.8.x?topic=models-autoai

- IBM Watson Machine Learning - Official Documentation

Link: https://www.ibm.com/products/watson-machine-learning

- Kaggle Data Set -https://www.kaggle.com/datasets/ziya07/power-system-faults-dataset

IBM CERTIFICATIONS

- Screenshot/ credly certificate(getting started with AI)


Getting Started with Artificial Intelligence

In recognition of the commitment to achieve professional excellence

Shreeniwas Pandit

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/ee800e66-a9f1-4a05-ac80-c2e3035400db

IBM

IBM CERTIFICATIONS

- Screenshot/ credly certificate(Journey to Cloud)

In recognition of the commitment to achieve professional excellence

Journey to Cloud: Envisioning Your Solution

IBM SkillsBuild

Shreeniwas Pandit

Has successfully satisfied the requirements for:

Journey to Cloud: Envisioning Your Solution

Issued on: Jul 17, 2025

Issued by: IBM SkillsBuild

Verify: https://www.credly.com/badges/e36aaeb3-fe7f-4816-86a0-f225fa4d7288

IBM

IBM CERTIFICATIONS

- Screenshot/ credly certificate(RAG Lab)

IBM **SkillsBuild** Completion Certificate

This certificate is presented to

Shreeniwas Pandit

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