CAPSTONE PROJECT

POWER SYSTEM FAULT DETECTION AND CLASSIFICATION

Presented By:

1. Srishti Tripathi- Shri Ramswaroop Memorial University- CSE

OUTLINE

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

PROBLEM STATEMENT

Example: Power distribution systems are prone to various faults due to environmental, equipment, or operational failures. These faults can lead to blackouts, equipment damage, or system instability. Current methods for fault detection are often slow, manual, or rule-based, lacking the intelligence for real-time classification. There is a critical need for automated and accurate fault detection and classification to ensure grid reliability, safety, and quick restoration.

PROPOSED SOLUTION

- Data Collection.
  - Real-time or historic phasor measurement data (voltage, current, angle).
  - Uploaded and organized in IBM Cloud Object Storage.
- Data Preprocessing (in Watson Studio)
  - Cleaning missing values, outliers.
  - Feature extraction (e.g., RMS, zero/positive/negative sequence components).
  - Scaling and normalization for ML compatibility.
  - Stored intermediate files in IBM Cloud Object Storage for reproducibility.
- Model Training (Watson Machine Learning)
  - Algorithms considered:
    - Random Forest Classifier
    - SVM
    - Logistic Regression
  - AutoAI : Automatically compares and selects the best-performing model.
  - Training and validation performed using cross-validation.
- Model Deployment
  - Deployed as REST API using Watson Machine Learning Service.
  - Webhook notifications or alerts can be generated using IBM Cloud Functions.

SYSTEM APPROACH

System Requirements -

Component	Description
Platform	IBM Cloud
Development Environment	IBM Watson Studio (Jupyter notebooks, AutoAI)
Model Deployment	IBM Watson Machine Learning
Storage	IBM Cloud Object Storage
Visualization	IBM Cognos Dashboard Embedded / Power BI
Minimum Hardware (Local)	Internet-enabled system with modern browser (Chrome/Edge)

SYSTEM APPROACH

Libraries & Tools Required -

Library / Tool	Purpose
pandas	Data manipulation and preprocessing
numpy	Numerical computations
matplotlib, seaborn	Data visualization
scikit-learn	Model training (Random Forest, SVM, metrics, train/test split)
xgboost	Gradient boosting-based fault classification
joblib	Model serialization and deployment
ibm_watson_machine_learning	Deploying models to Watson Machine Learning as API endpoints
AutoAI (IBM service)	Automated model selection and hyperparameter tuning

ALGORITHM & DEPLOYMENT

- **Algorithm Selection:**

  The Random Forest Classifier was chosen for this project due to its robustness, ability to handle both numerical and categorical variables, and strong performance on multi-class classification problems. Given the variety of fault types (e.g., Line Breakage, Transformer Failure, Overheating), and the mix of features such as Voltage, Current, Weather Condition, and Maintenance Status, Random Forest proved effective in capturing complex relationships and reducing overfitting.

- **Data Input:**

Feature	Description
Voltage (V)	Voltage measured at the fault location
Current (A)	Fault current level
Power Load (MW)	Electrical load during the event
Temperature (°C)	Ambient temperature at fault time
Wind Speed (km/h)	Environmental impact factor
Weather Condition	Categorical (Clear, Rainy, Snowy, Windstorm)
Maintenance Status	Status of maintenance before the fault
Component Health	Health state of the affected component
Duration of Fault (hrs)	How long the fault persisted
Down Time (hrs)	System outage duration caused by the fault

Target Variable: Fault Type

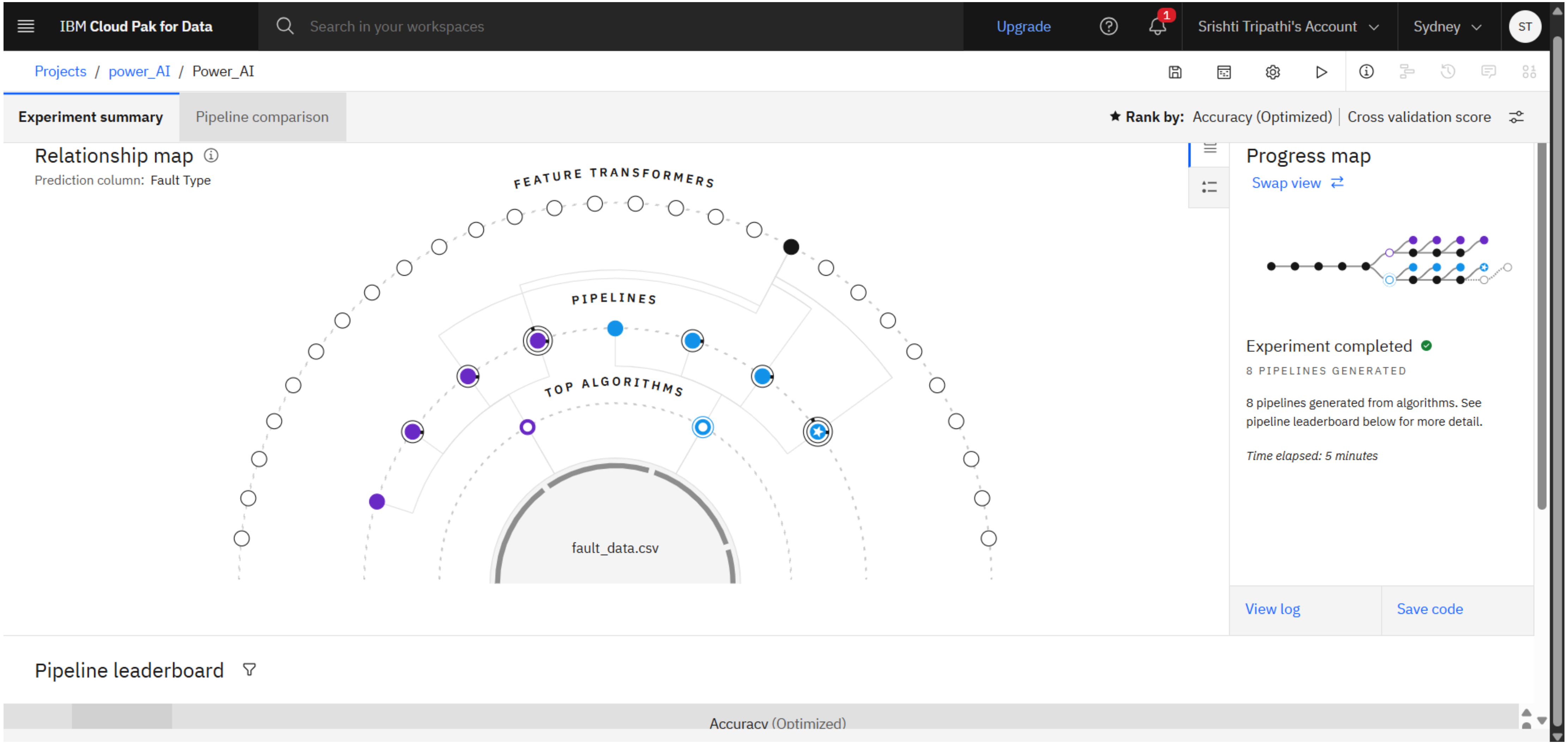
Categorical variables were label encoded or one-hot encoded during preprocessing.

edunet foundation

ALGORITHM & DEPLOYMENT

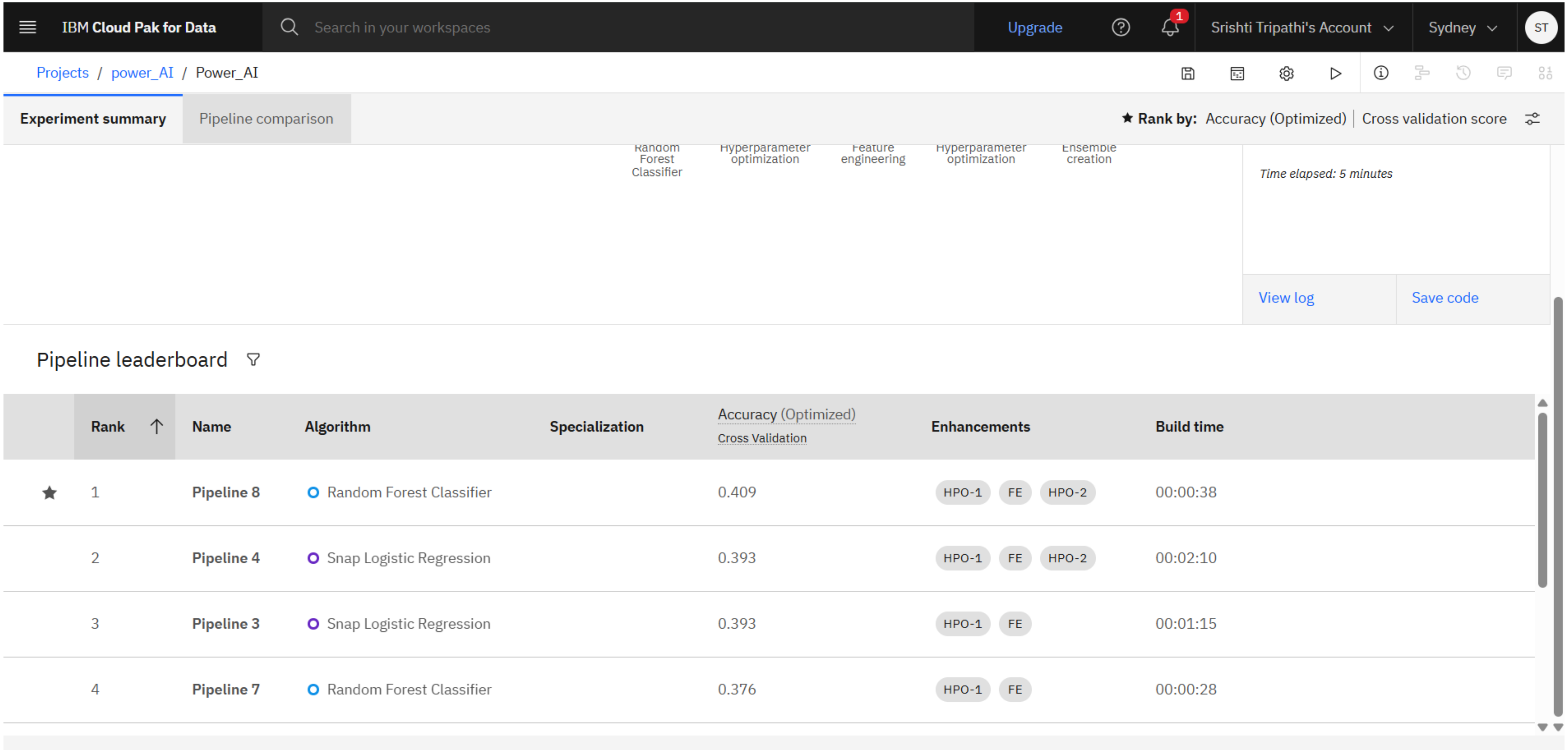
- Training Process:
  - Preprocessing (in IBM Watson Studio):
    - Handled missing values and inconsistent entries.
    - Encoded categorical features (Weather Condition, Maintenance Status, etc.).
    - Scaled numerical features.
  - Model Training:
    - Split the dataset into 80% training and 20% testing.
    - Used cross-validation to ensure generalization.
    - Performed grid search for tuning parameters like number of trees and max depth.
  - Performance Metrics:
    - Accuracy: % of correctly classified fault types
    - Precision & Recall: For each fault category
    - Confusion Matrix: Visual breakdown of prediction success per class
- Prediction Process:
  - Input: Real-time or batch input of feature data via API (voltage, current, weather, etc.)
  - Model Inference: WML predicts the most probable fault type.
  - Output: Predicted fault label + confidence score (e.g., “Transformer Failure – 87%”).
  - Visualization: Output logged and displayed on Cognos dashboard for operator insights.

RESULT

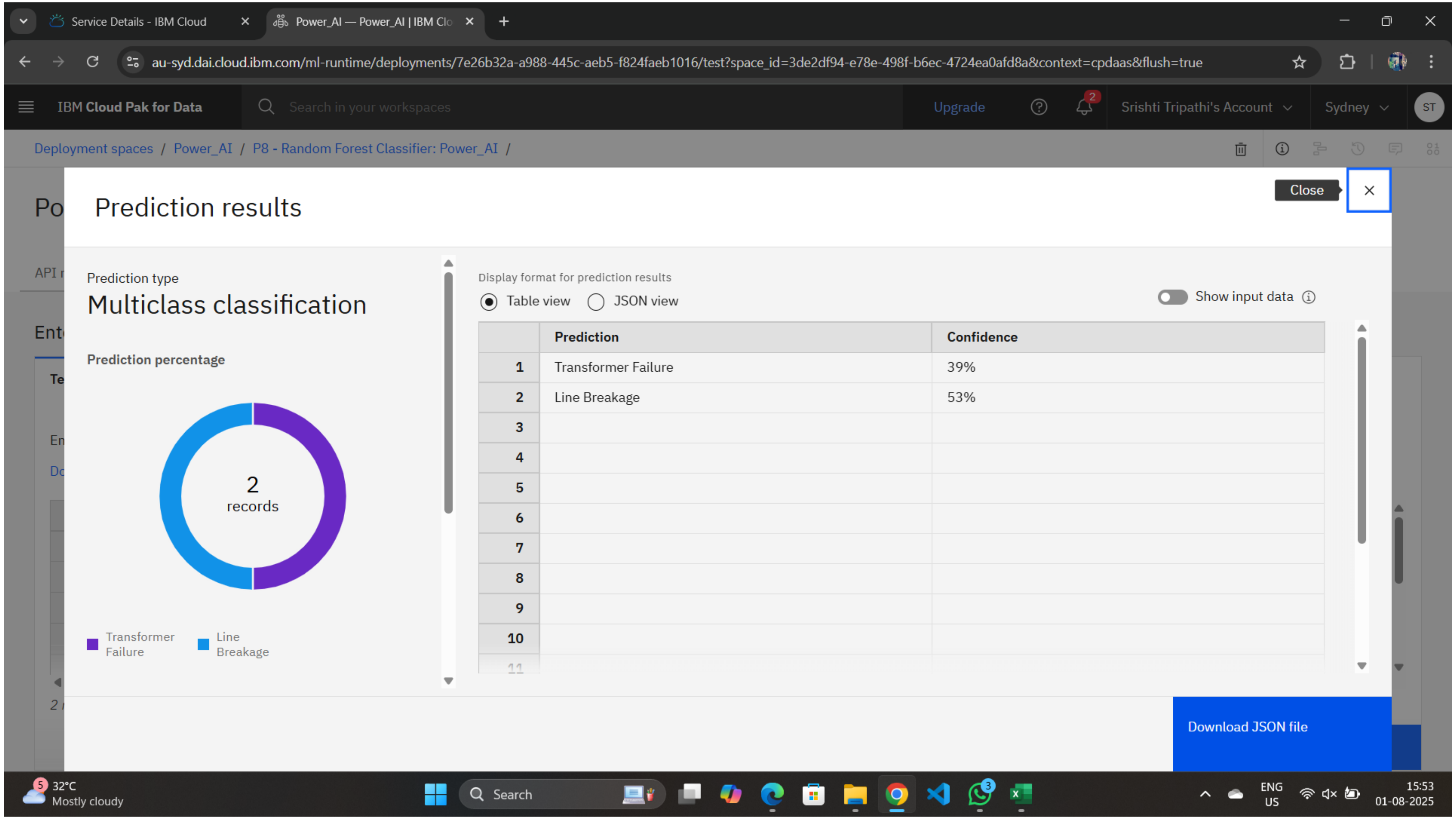


edunet
foundation

RESULT


IBM Cloud Pak for Data
Search in your workspaces
Upgrade
Srishti Tripathi's Account
Sydney
ST
Projects / power_AI / Power_AI
Experiment summary
Pipeline comparison
Rank by: Accuracy (Optimized) | Cross validation score
Random Forest Classifier
Hyperparameter optimization
Feature engineering
Hyperparameter optimization
Ensemble creation
Time elapsed: 5 minutes
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 00:00:38
2 Pipeline 4 Snap Logistic Regression 0.393 HPO-1 FE HPO-2 00:02:10
3 Pipeline 3 Snap Logistic Regression 0.393 HPO-1 FE 00:01:15
4 Pipeline 7 Random Forest Classifier 0.376 HPO-1 FE 00:00:28

RESULT



IBM Watson Machine Learning Multiclass Classification Result

CONCLUSION

This project successfully demonstrates the use of IBM Cloud-based AI/ML tools for detecting and classifying faults in power distribution systems. By leveraging services like Watson Studio, Watson Machine Learning, and Cloud Object Storage, the system provides accurate, real-time fault classification for conditions.

The solution is scalable, automated, and supports easy deployment and visualization. Overall, it highlights the power of cloud-based machine learning in improving grid reliability and fault response times, paving the way for smarter and more resilient energy systems.

FUTURE SCOPE

- Edge deployment with IBM Edge Application Manager
- Real-time analytics using Watson IoT
- Predictive maintenance integration with IBM Maximo
- Fault location prediction using deeper ML/DL models

REFERENCES

- IBM Cloud documentation
- Watson Studio & AutoAI resources
- IEEE papers on power system faults
- Tutorials from IBM Developer & Cloud Pak for Data labs

IBM CERTIFICATIONS

In recognition of the commitment to achieve professional excellence

Getting Started with Artificial Intelligence
IBM SkillsBuild

Srishti Tripathi

Has successfully satisfied the requirements for:

Getting Started with Artificial Intelligence

Issued on: Jul 16, 2025
Issued by: IBM SkillsBuild

Verify: https://www.credly.com/badges/ef726628-827e-4905-814e-f5ecbf83e5c0

IBM

IBM CERTIFICATIONS

In recognition of the commitment to achieve professional excellence

Srishti Tripathi

Has successfully satisfied the requirements for:

Journey to Cloud: Envisioning Your Solution

Issued on: Jul 16, 2025

Issued by: IBM SkillsBuild

Verify: https://www.credly.com/badges/16882f46-a82d-4abc-be4e-27a5fc7773ae

IBM

IBM CERTIFICATIONS

7/17/25, 1:31 PM Completion Certificate | SkillsBuild

IBM **SkillsBuild** Completion Certificate

This certificate is presented to

Srishti Tripathi

for the completion of

Lab: Retrieval Augmented Generation with LangChain

(ALM-COURSE_3824998)

According to the Adobe Learning Manager system of record

Completion date: 17 Jul 2025 (GMT) **Learning hours:** 20 mins

THANK YOU

edunet
foundation